Burroughs

Reference Manual

# B 1000 Systems

# Supervisory Message Control System (SMCS)

(Relative to the SMCS 10.0.007 Release)

Priced Item
Printed in U.S.A.
May 1983

1108891

Burroughs

*Reference Manual*

# B 1000 Systems Supervisory Message Control System (SMCS)

*(Relative to the SMCS 10.0.007 Release)*

*Copyright © 1983, Burroughs Corporation, Detroit, Michigan 48232*

*Priced Item*
*Printed in U.S.A.*
*May 1983*

*1108891*

Burroughs cannot accept any financial or other responsibilities that may be the result of your use of this information or software material, including direct, indirect, special or consequential damages. There are no warranties extended or granted by this document or software material.

You should be very careful to ensure that the use of this software material and/or information complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Correspondence regarding this publication should be forwarded using the Remarks form at the back of the manual, or may be addressed directly to TIO West Documentation, Burroughs Corporation, 1300 John Reed Court, City of Industry, California 91745, U.S.A.

# LIST OF EFFECTIVE PAGES

| Page | Issue |
|---|---|
| Title | Original |
| ii | Original |
| iii | Original |
| iv | Blank |
| v thru ix | Original |
| x | Blank |
| 1-1 thru 1-2 | Original |
| 2-1 thru 2-5 | Original |
| 2-6 | Blank |
| 3-1 thru 3-3 | Original |
| 3-4 | Blank |
| 4-1 thru 4-12 | Original |
| 5-1 thru 5-57 | Original |
| 5-58 | Blank |
| 6-1 thru 6-9 | Original |
| 6-10 | Blank |
| 7-1 thru 7-17 | Original |
| 7-18 | Blank |
| A-1 thru A-8 | Original |
| B-1 | Original |
| B-2 | Blank |
| C-1 thru C-2 | Original |
| D-1 | Original |
| D-2 | Blank |
| E-1 | Original |
| E-2 | Blank |
| F-1 thru F-3 | Original |
| F-4 | Blank |
| G-1 thru G-7 | Original |
| G-8 | Blank |
| H-1 thru H-3 | Original |
| H-4 | Blank |
| 1 thru 3 | Original |
| 4 | Blank |

# TABLE OF CONTENTS


| Section | Title | Page |
|---|---|---|
| 1 | INTRODUCTION | 1-1 |
| | General | 1-1 |
| | Document Overview | 1-1 |
| | Related Documents | 1-2 |
| 2 | SMCS FEATURES AND CAPABILITIES | 2-1 |
| | General | 2-1 |
| | Hardware Requirements | 2-2 |
| | Memory Requirements | 2-2 |
| | Software Requirements | 2-3 |
| | Operating System (MCPII) | 2-3 |
| | Network Controller | 2-3 |
| | SYSTEM/MAKEUSER | 2-3 |
| | DMPALL | 2-3 |
| | Software Options | 2-4 |
| | Burroughs Network Architecture (BNA) | 2-4 |
| | CANDE | 2-4 |
| | RD | 2-4 |
| | SYCOM | 2-5 |
| | TEXT/EDITOR | 2-5 |
| | User Program | 2-5 |
| | SMCS Program Capabilities | 2-5 |
| 3 | SMCS FUNCTIONAL DESCRIPTION | 3-1 |
| | General | 3-1 |
| | MCP Message Routing | 3-1 |
| | Recognition of Attached Stations | 3-1 |
| | Open Approval | 3-1 |
| | SMCS is the Parent Job | 3-1 |
| | SMCS is not the Parent Job | 3-3 |
| 4 | SMCS INSTALLATION | 4-1 |
| | General | 4-1 |
| | SMCS Program Switches | 4-1 |
| | JOBS File | 4-1 |
| | JOBS File Syntax | 4-2 |
| | Creation of the JOBS File | 4-6 |
| | (SYSTEM)/USERCODE File | 4-6 |
| | Program Priorities | 4-6 |

# TABLE OF CONTENTS (Cont)


| Section | Title | Page |
|---|---|---|
| | Generating the Network Controller | 4-6 |
| | Network Controller Name | 4-6 |
| | Declaration Section | 4-6 |
| | Request Section | 4-7 |
| | Control Section | 4-7 |
| | Terminal Section | 4-7 |
| | Station Section | 4-7 |
| | Line Section | 4-9 |
| | File Section | 4-9 |
| | Network Controller Example | 4-9 |
| | Setup of the MCP Name Table | 4-11 |
| | Network Controller | 4-11 |
| | SMCS | 4-11 |
| | Workfile Naming Conventions | 4-12 |
| | SMCS OPERATING INSTRUCTIONS | 5-1 |
| | SMCS Program Execution | 5-1 |
| | SMCS Program Termination | 5-1 |
| | SMCS Input Commands | 5-1 |
| | ATTACH | 5-3 |
| | BROADCAST | 5-5 |
| | BYE | 5-6 |
| | CHANGE | 5-7 |
| | CLEAR | 5-10 |
| | CLOSE Command | 5-11 |
| | CONTINUE Command | 5-12 |
| | DETACH | 5-13 |
| | DUMP | 5-15 |
| | DYNAMIC | 5-16 |
| | FILE | 5-17 |
| | HARDWARE | 5-18 |
| | HELP | 5-19 |
| | ID | 5-20 |
| | INITIATE | 5-21 |
| | JOBS | 5-22 |
| | LOGON Command | 5-23 |
| | MACRO | 5-25 |
| | MAIL | 5-27 |
| | MAKE | 5-28 |
| | MOVE | 5-31 |
| | NEWS | 5-33 |
| | PASS | 5-34 |
| | PASSWORD | 5-35 |
| | READY | 5-36 |
| | REMOVE | 5-37 |
| | REPORT | 5-38 |

# TABLE OF CONTENTS (Cont)


| Section | Title | Page |
|---|---|---|
| | SEND | 5-40 |
| | SIGN ON | 5-42 |
| | SIGN OFF | 5-44 |
| | SIGNAL | 5-45 |
| | STATUS | 5-46 |
| | STOP | 5-48 |
| | SYSTEM | 5-49 |
| | TABS | 5-50 |
| | TERMINATE Command | 5-52 |
| | TRACE | 5-53 |
| | TRANSLATE | 5-54 |
| | USER | 5-55 |
| | ZIP | 5-56 |
| | ? <MCP control instruction> | 5-57 |
| 6 | MACRO PROCEDURES | 6-1 |
| | Macro Procedure Creation | 6-1 |
| | Macro Procedure Invocation | 6-1 |
| | Step Macro Procedures | 6-4 |
| | Step Macro Commands | 6-5 |
| | Limits | 6-9 |
| 7 | SYSTEM OUTPUT MESSAGES | 7-1 |
| | Master Control Program Messages | 7-1 |
| | Network Controller Messages | 7-1 |
| | SMCS Output Messages | 7-1 |
| | Informational Messages | 7-1 |
| | Warning Messages | 7-1 |
| | Error Messages | 7-1 |
| | Fatal Error Message | 7-2 |
| | Output Messages | 7-3 |
| A | GLOSSARY OF TERMS | A-1 |
| B | CHARACTER SET | B-1 |
| C | RESTRICTED MCP COMMANDS | C-1 |
| D | SUMMARY OF FILES | D-1 |
| | SMCS File Information | D-1 |
| E | THE MAIL FILE | E-1 |
| F | THE TRACE FILE | F-1 |
| | Record Format | F-1 |
| | IQUE SAVE | F-1 |
| | JOBQ IN | F-1 |
| | MCPQ IN | F-1 |
| | MSGQ CHIN | F-2 |
| | MSGQ IN | F-2 |
| | MSGQ SAVE | F-2 |
| | REMOTE IN | F-2 |
| | REMOTE OUT | F-2 |

# TABLE OF CONTENTS (Cont)


| Section | Title | Page |
|---|---|---|
| | RF INFO-2 | F-2 |
| | SCHQ FIRE | F-3 |
| | SCHQ IN | F-3 |
| | ODT OUT | F-3 |
| | ZIP | F-3 |
| G | SMCS PROGRAM-PASS MESSAGES | G-1 |
| | General | G-1 |
| | Program-Pass Output Messages | G-1 |
| | Message Header Format | G-2 |
| | Remote Key Format | G-2 |
| | Text Format | G-3 |
| | Program-Pass Input Messages | G-5 |
| | Message Header Format | G-5 |
| | Remote Key Format | G-5 |
| | Text Format | G-5 |
| H | SYNTAX AND NOTATION CONVENTIONS | H-1 |
| | Required Items | H-1 |
| | Optional Items | H-1 |
| | Loops | H-1 |
| | Bridges | H-3 |

# LIST OF ILLUSTRATIONS


| Figure | Title | Page |
|---|---|---|
| 2-1 | SMCS Basic Message Flow | 2-2 |


# LIST OF TABLES


| Table | Title | Page |
|---|---|---|
| 5-1 | SMCS Command Summary | 5-2 |
| D-1 | Summary of Files | D-1 |
| G-1 | Program-Pass Message Header Format | G-2 |
| G-2 | Program-Pass Remote Key Format | G-2 |
| G-3 | Program-Pass Output Message Text Format | G-3 |
| G-4 | Program-Pass Output Text Format | G-5 |

# SECTION 1
# INTRODUCTION

## GENERAL

The Supervisory Message Control System (SMCS) is described in this document, and the commands and the general operation of the system and its associated stations are defined. The SMCS program is for users of interactive data communications systems and offers the capability of a message control system interface to the data communications network, thereby greatly expanding the functional flexibility of the stations.

## DOCUMENT OVERVIEW

This document is intended for individuals with a working knowledge of the basic hardware and software functions of B 1000 data processing systems. A brief description of each section and appendix follows:

Section 1: Introduction
Provides a brief introduction to the SMCS program, related documentation, and notation conventions.

Section 2: SMCS Features And Capabilities
Describes the features and capabilities of the SMCS program for data processing managers.

Section 3: SMCS Functional Description
Describes the functions and system requirements, of the SMCS program for systems programmers.

Section 4: SMCS Installation
Describes the generation of the network controller, the creation of the JOBS file, and the SMCS program switches. This section is intended for the operations manager or any person who installs the SMCS program.

Section 5: SMCS Operating Instructions
Describes the SMCS program operating instructions and the SMCS input commands. This section is intended for the operators of the SMCS program.

Section 6: Macro Procedures
Describes the creation and invocation of macro procedures. This section is intended for systems programmers.

Section 7: System Output Messages
Describes the MCP, network controller, and SMCS program output messages. This section is intended for the operators of the SMCS program.

Appendix A: Glossary Of Terms
Contains a glossary of terms used in this document.

Appendix B: Character Set
Describes the character set used by the SMCS program.

Appendix C: Restricted MCP Commands
Describes the MCP commands allowed at work stations.

Appendix D: Summary of Files
Provides a summary of the SMCS program files.

Appendix E: The MAIL File
Describes the format of the MAIL file.

Appendix F: The TRACE File
Describes the format and use of the TRACE file.

Appendix G: SMCS Program-Pass Messages
Describes the format and use of the SMCS program-pass messages.

Appendix H: Syntax and Notation Conventions
Describes the syntax and notation conventions used in this document.

## RELATED DOCUMENTS

The following documents are referenced in this document:

B 1000 Systems System Software Operation Guide, Volume 1, form number 1108982.

B 1000 Systems System Software Operation Guide, Volume 2, form number 1138542.

B 1000 Systems Network Definition Language (NDL) Reference Manual, form number 1073715.

B 1000 Systems Command AND Edit (CANDE) User's Manual, form number 1090586.

B 1000 System Communications Module (SYCOM) Reference Manual, form number 1108859.

B 1000 Systems Burroughs Network Architecture (BNA) Installation and Operations Reference Manual, form number 1127578.

B 1000 Systems Remote Display (RD) Reference Manual, form number 1108933.

B 1800/B 1700 TEXT/EDITOR Reference Manual, form number 1090610.

TD 730/TD 830 System Reference Manual, form number 1093738.

MT 983/MT 985/MT 993 Display Terminals Reference Manual, form number 1118361.

# SECTION 2
# SMCS FEATURES AND CAPABILITIES

## GENERAL

The SMCS program provides Message Control System (MCS) interface to the data communications network. The SMCS system software consists of the SMCS program and the network controller.

The SMCS program supervises a data communications system and provides functions such as the following:

1. Controls security through a log-on procedure, checking user security codes and granting access to user files.
2. Approves or denies the opening of all remote files involving stations under its control. A station is a physical terminal that is designed to both receive and/or transmit in an environment associated with the task to be performed. Once the station or stations are attached to a remote file, the SMCS program does not participate in the input/output message flow between the application program and the station or stations.
3. Allows communication from one station to another station, from a station to the Operator Display Terminal (ODT), and from the ODT to a station.
4. Designates stations as workstations that have limited access to MCP commands.
5. Designates stations as remote ODT stations that have access to MCP commands and information about the status of the system.
6. Designates stations as controller stations that can perform certain network reconfiguration operation commands from that station.
7. Allows a user at a remote station, through the Program-pass interface, to request a function of a program in the local system without signing on to the program.
8. Allows expanded network reporting, tailored to individual remote stations for those stations designated as restricted. Expanded network reporting is also possible for the whole network and the operating system for those stations designated as unrestricted.
9. Initiates system tracing, from controller stations or the system ODT, to take advantage of the debugging facilities provided by the B 1000 operating system.
10. Allows the user to create macro procedures that invoke MCP control strings or SMCS program commands without the need to enter lengthy control strings.

To use the SMCS system software, both the SMCS program and the network controller must be executing. The network controller is a data communications operating system and is responsible for handling all of the details of the data communication line discipline. Communication is achieved using the remote file mechanism. Typically, message flow is handled in the following way:

1. In response to a poll from the network controller, a station transmits a message to the network controller.
2. Upon receipt of the message, the network controller writes it into the input queue of the appropriate program.
3. The program reads the message, processes the message, and generates a reply to the original message.
4. The program then sends the reply to the station by writing it into the input queue of the network controller.
5. Upon receiving the reply in the input queue, the network controller initiates the appropriate procedures to transmit the reply to the designated station.

Figure 2-1 illustrates in block form the basic message flow in the Supervisory Message Control System.

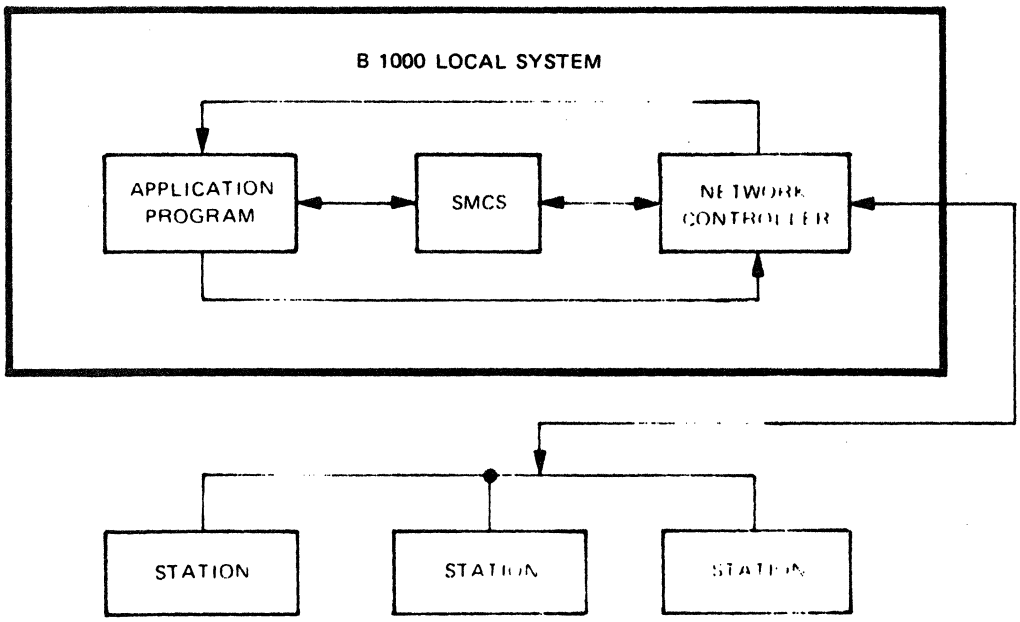



**Figure 2-1. SMCS Basic Message Flow**

# HARDWARE REQUIREMENTS

The following hardware elements are required to execute the B 1000 SMCS program:

B 1900, B 1800, or B 1700 processor

Multiline control or single-line control

Line adapters as required

Disk storage

Operator Display Terminal (ODT)

Terminal Devices:
- MT 983 Teletype
- MT 985
- MT 993
- TD 830
- TD 820

# MEMORY REQUIREMENTS

The memory requirement for the Supervisory Message Control System (SMCS) is 25.0 K bytes.

The required memory for the SMCS program is only utilized when a SMCS command is in process. Since the SMCS program is a non-participating Message Control System (MCS), it does not participate in the input/output message flow between an application program and the station once the station is signed on to a program with the exception of program-pass messages.

# SOFTWARE REQUIREMENTS

The following software elements are required to execute the SMCS program:

Operating System (MCPII)

Network Controller

SYSTEM/MAKEUSER Program

DMPALL Program

## Operating System (MCPII)

The Master Control Program (MCP) is an operating system that manages the demands and resources of a B 1000 system. It reduces the programming effort by providing commonly needed functions and services. The MCP requires strict adherence to program structures and procedures to ensure communication between different programs, interpreters, and the MCP itself.

The primary function of the MCP is to optimize the productivity of a system. External intervention is held to a minimum, and maximum throughput is achieved by incorporating into the MCP the primary tasks of input/output control, file handling, multiprogramming, system interrupts, memory allocation, and operator interfacing.

## Network Controller

The network controller is generated by the Network Definition Language (NDL) compiler. The network controller processes and supervises the flow of messages between application programs and the remote network (stations). It performs all data communication line discipline functions necessary for proper line/terminal control. The network controller handles the initiation and completion of data communication I/O and services exception conditions. The network controller is configured by symbolic specifications to the NDL compiler and these specifications are specific to the configuration of the system on which it is to be run.

## SYSTEM/MAKEUSER

The SYSTEM/MAKEUSER program is a utility program that creates, accesses, or modifies the system usercode file of allowable usercode and password combinations. A list of all valid usercode/password combinations is maintained in this file, which is named (SYSTEM)/USERCODE.

The usercode/password combinations are used to implement file security to control and/or restrict usage of files and/or programs.

## DMPALL

The DMPALL program is a general media conversion program that allows conversion of files from one medium to another. It also lists files from any medium with byte, digit, or combined format. The DMPALL program can be used to create the JOBS file and to print specific SMCS files such as the JOBS file, the MAIL file, and the TRACE file.

# SOFTWARE OPTIONS

The following software elements are optional but can be used in conjunction with the SMCS program to serve the particular needs of the user:

Burroughs Network Architecture (BNA)

CANDE

RD

SYCOM

TEXT/EDITOR

User Program

## Burroughs Network Architecture (BNA)

Burroughs Network Architecture (BNA) provides a means to perform distributed data processing such as remote data base access, file transfer, remote task initiation and control, logical I/O, inter-process communication, and resource sharing.

BNA is segmented into two major subdivisions: Host Services and Network Services.

Host Services performs such functions as operator inquiry, job transfer, task initiation and control, logical I/O, inter-process communication, terminal transfer. Host Services also provides a resource environment definition.

Network Services provides the communication protocols and algorithms required to support the BNA environment. The protocols and algorithms make provisions for the routing of messages throughout the network, message segmentation and reassembly, end-to-end message integrity, network reconfiguration, and interconnecting hosts by means of data communications links.

Refer to the B 1000 System Burroughs Network Architecture (BNA) Installation and Operations Reference Manual for more information on the BNA system.

## CANDE

The B 1000 CANDE program provides generalized file preparation and updating capabilities in an interactive, terminal-oriented environment.

Refer to the B 1000 Systems Command AND Edit (CANDE) User's Manual for more information on the CANDE program.

## RD

The Remote Display (RD) program provides users of a remote display terminal with the ability to obtain system status information. The system status information includes file and program attributes, the current system configuration, and job information.

Refer to the B 1000 Systems Remote Display (RD) Reference Manual for more information on the RD program.

### SYCOM

The SYCOM program provides users with the ability to communicate with remote Burroughs systems, to transfer files between systems, to execute jobs on a remote system, and for a terminal on the local system to function as a terminal on a remote system.

Refer to the B 1000 System Communcations Module (SYCOM) Reference Manual for more information on the SYCOM program.

### TEXT/EDITOR

The TEXT/EDITOR program provides users with the ability to get data into or out of the system. It makes possible the creation or updating of disk files of sequenced card images from a remote terminal.

Refer to the B 1800/B 1700 TEXT/EDITOR Reference Manual for more information on the TEXT/EDITOR program.

### User Program

A user can develop an interactive program tailored to the specific needs of an installation. These specific needs include application functions such as data entry or information retrieval.

## SMCS PROGRAM CAPABILITIES

Since the SMCS program requires that the network controller be executing, any maximum capacity that applies to the network controller also applies to the SMCS program. The capabilities of the SMCS program are:

1. May contain up to 128 stations.
2. May have up to 64 programs spawned by the SMCS program as a result of a SIGN ON or a PASS command. Also included are programs, which open a remote file, that are initiated at a remote station.
3. May have up to 64 remote files opened by programs spawned by the SMCS program.
4. The maximum message text size is 1920 bytes.
5. May have up to 6 RECALL commands in progress at any one time.
6. May have up to 8192 entires in the MAIL file.

# SECTION 3
# SMCS FUNCTIONAL DESCRIPTION

## GENERAL

This section describes the MCP message routing system, the recognition of attached stations, and the Open approval mechanisms,

## MCP MESSAGE ROUTING

All MCP control strings are zipped with a session number equal to the Logical Station Number (LSN) of the station that initiated the zip.

The SMCS program uses a combination of the session number, which is equal to the LSN, and the usercode index (if there is one) to deliver incoming messages. The following is the order in which the SMCS attempts to route a message from the MCP to a station.

1. If the message is received without a usercode index, the SMCS program delivers the message to the LSN that initiated the ZIP command.
2. If the message is received with a usercode index, the SMCS program delivers the message to the LSN that initiated the ZIP command if 1) the usercode index associated with that LSN is the same as the usercode index in the message or, if 2) that LSN has no usercode index associated with it.
3. If point 2 is not satisfied, the SMCS program searches its station table looking for an LSN that is logged on with the same usercode index as in the message. If a match is found, the SMCS program delivers the message to the first station that it finds logged on with the usercode index in the message.
4. If the above conditions are not satisfied, the message is placed in the MAIL file.

## RECOGNITION OF ATTACHED STATIONS

The SMCS program automatically recognizes new stations attached to its remote file. This feature enables the SMCS program to run subordinate to any other MCS, such as a GEMCOS MCS, that would attach stations to the SMCS remote file.

## OPEN APPROVAL

Two aspects of Open approval are discussed: (1) SMCS is the parent job and (2) SMCS is not the parent job.

The reader should refer to the B 1000 Systems Network Definition Language (NDL) Reference Manual for a general explanation of the Open approval mechanism.

### SMCS is the Parent Job

The SMCS program is the parent job for all jobs (or offspring) that it zip executes (or spawns).

The SMCS program receives the Open request from the network controller for all jobs it spawns unless all of the stations in the Open station list belong to a different MCS.

If a program that was spawned by the SMCS program spawns a second program that opens a remote file, the SMCS program is considered to be the parent job for the second program if the first program goes to end of job. The SMCS program can subsequently receive an Open request for the second program unless all of the stations in the Open station list belong to a different MCS.

The SMCS program uses the following criteria when approving or denying an Open request for which it is the parent job:

1. If the program is in the JOBS file and was initiated through a SIGN ON command, only stations entering the SIGN ON command are approved for the Open request unless program access is restricted by the USERCODE or USERFILE options. If a station list is included in the Open request, it is ignored by the SMCS program. Refer to the discussion on the JOBS file in section 4, SMCS Installation, for more information on the setup of the JOBS file.
2. If the program was initiated as a result of an MCP control string (EXECUTE <program>), the SMCS program uses the station list in an Open request if all the stations in the Open station list are not controlled by the SMCS program. If any of the stations in the Open station list are controlled by the SMCS program or if there is no station list, the initiating station is approved in the Open request.

   This feature allows a program to direct output to a terminal other than the one it was initiated on, without the program being a Message Control System (MCS) or having two remote files in the program. The terminal to which output is directed by the program cannot be controlled by the SMCS program. For example, this feature can be used to list output on a remote printer, such as an AP 300 series remote printer.

3. If a program spawned by the SMCS program spawns a second program that opens a remote file, then the SMCS program is considered to be the parent job for the second program if the first program has gone to end of job before the second program opens its remote file. The network controller forwards the Open request to the parent job, that is the SMCS program, unless all of the stations in the station list are controlled by a different MCS. If the SMCS program receives an Open request, the following steps are taken:

   1) If there is a station list in the Open request and all the stations in the station list are not controlled by the SMCS program, then the SMCS program uses the station list in approving the Open request.
   2) If any of the stations in the station list are controlled by the SMCS program or there is no station list in the Open request for the second program, then a check is made to determine how the first program was spawned by the SMCS program. If the first program was spawned as a result of a SIGN ON, PASS, or INITIATE command or the AUTO-START option being specified in the SMCS/JOBS file, then the Open request for the second program is denied. If the first program was spawned as a result of an MCP control instruction (EXECUTE <program>), the SMCS program attempts to approve the Open request for the Logical Station Number (LSN) which initiated the first program. If that LSN cannot be approved in the Open request for the second program, the Open request is denied.

4. If the Open request is for the second input or input/output remote file Open, then the SMCS program approves the Open using the Open station list if all of the stations in the Open station list are not controlled by the SMCS program.

## SMCS is not the Parent Job

The SMCS program receives Open requests for programs which were executed at the system ODT if:

1. All of the stations in the Open station list belong to the SMCS program, or
2. Some of the stations in the Open station list belong to the SMCS program and the rest are not assigned to a remote file.

The SMCS program receives Open requests for programs that were spawned by another MCS if all of the stations in the Open station list belong to the SMCS program.

The SMCS program uses the following criteria when approving or denying Open requests for programs for which it is not the parent job:

1. The SMCS approves an Open request for those stations in the Open station list that

   1) are not attached to another remote file or are currently attached to another remote file and have a remote file Open type of output-only.
   2) have a MYUSE attribute matching the Open type. For example, the remote file Open type is input and the MYUSE attribute of the station is not output-only.
   3) are not logged on to the SMCS program with a USER or ID command.

2. If the Open request is the second input or input/output remote file Open request, the Open request is approved if all of the stations in the Open station list are not controlled by the SMCS program.

# SECTION 4
# SMCS INSTALLATION

## GENERAL

Before the SMCS program can be executed, the following steps must be performed to install the SMCS program and associated system software.

1. Set or reset the SMCS program switches.
2. Prepare the JOBS file.
3. Create the (SYSTEM)/USERCODE file.
4. Assign program priorities.
5. Compile the network controller.
6. Set up the MCP name table.

## SMCS PROGRAM SWITCHES

The following program switches are used by the SMCS program:

Program Switch 0

If program switch 0 = 0 at execution time, disk tracing is performed. The trace information is written to a circular disk file, labeled TRACE, that contains 500 records. If program switch 0 = 1 at execution time, printer tracing is performed. The trace information is written to the line printer. If tracing is not desired, it is necessary to set program switch 0 to a value greater than 1. Refer to appendix F, The TRACE file, for more information on the SMCS trace function. The default value for program switch 0 is 0 (disk tracing).

Program Switch 2

Program switch 2 suppresses the automatic execution at SMCS beginning of job for any job that was in the JOBS file with the AUTO-START option specified. If program switch 2 is set, the AUTO-START option in the JOBS file for all jobs is ignored. The default value for program switch 2 is 0.

Program Switch 3

Program switch 3 enables the macro invocation functions of the SMCS program. If program switch 3 is set at beginning of job, the macro procedure invocation capabilities are enabled and the DIRECTORY file is opened or created if it is not already resident on disk. The default value for program switch 3 is 0.

If program switch 3 is reset at begining of job and macro procedures are desired, then the MACRO ON command must be entered from a controller station or from the system ODT. Refer to section 5, SMCS Operating Instructions for more information on the MACRO command.

## JOBS FILE

All program-identifiers specified in the SIGN ON, PASS, or INITIATE commands must be specified in the JOBS file for the SMCS program. This file contains data concerning a particular job to be run in conjunction with the SMCS program. At beginning of job of the SMCS program and when the first SIGN ON, PASS, or INITIATE command is received for a program, the SMCS program checks the JOBS file for the program identifier specified in the command for special instructions on how to execute the job.

The SMCS program has the capability to restrict program access by the SIGN ON and PASS commands to certain usercodes or partial usercodes. This is done by specifying the USERCODE and USERFILE options.

The SMCS program allows a timer value to be specified in the NO-EOF option. If a timer value is specified, the SMCS program waits for the specified amount of time before sending an EOF branch to the program when all signed-on stations have signed off the program and all stations that have passed to the program have logged off the SMCS program.

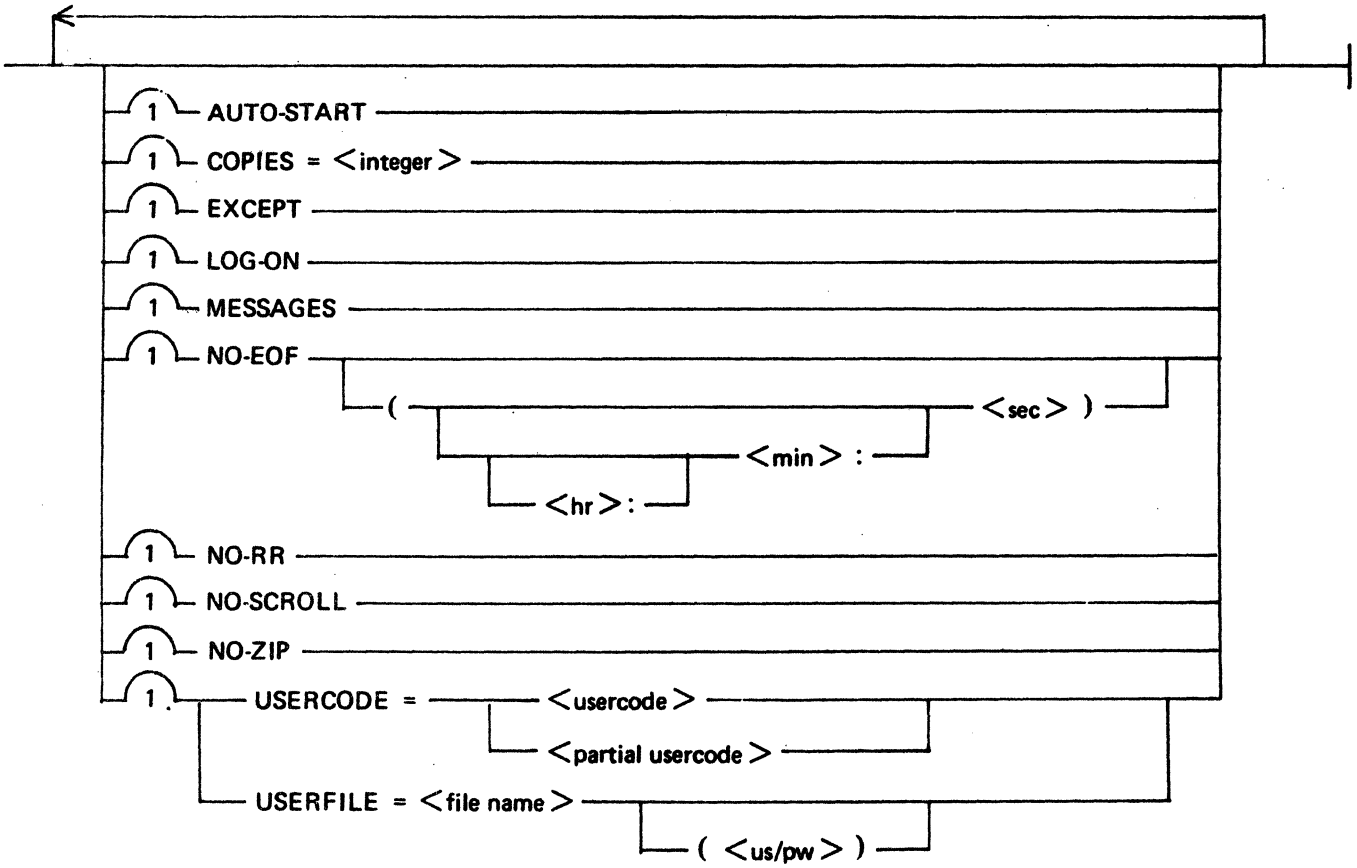
During the run of the SMCS program, the JOBS file remains open and a table of programs executed or already executing from the JOBS file is maintained. This table contains the program identifiers and pointers into the JOBS file. If the user needs to change the JOBS file, the CLOSE command closes the JOBS file. The JOBS file remains closed until the SMCS program needs to access it again.

Comments in the JOBS file are delimited by the percent sign character (%).

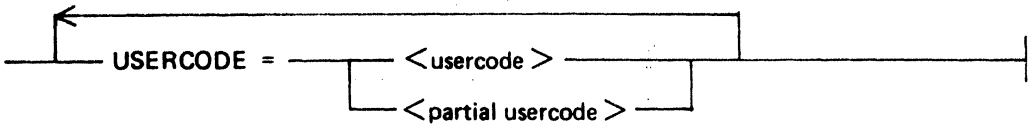
## JOBS File Syntax

$ <program-id> <option list> ; <zip string>

The syntax for the <option list> is as follows:



The syntax for a USERFILE record is:




Semantics

<program-id>
The <program-id> is the name by which the program is to be known to the SMCS program. The <program-id> must be 32 characters or less and can be different from the actual name of the program.

The <program-id> for the RD program must be the token RD in order for the JOBS, MIX, and WY commands to be automatically forwarded to the RD program.

AUTO-START
If the AUTO-START option is specified, the <program-id> is spawned at the beginning of job of the SMCS program.

COPIES
The COPIES option specifies the maximum number of copies of <program-id> that can be spawned by the SMCS program.

EXCEPT
If the EXCEPT option is specified, an exception indicator is sent to the program when the station signs off using the *SIGN OFF command.

LOG-ON
If the LOG-ON option is specified, the usercode and password used to logged on to the SMCS program with, using the USER command, is sent as the first input to <program-id> for log on purposes. If there was any data string included in the SIGN ON command, that string is sent as second input to the program.

MESSAGES
The MESSAGES option causes all MCP output messages for <program-id> to be delivered to all stations that are signed on to <program-id>.

If the MESSAGES option is specified, the SMCS program sets the LS boolean when it spawns <program-id>. This means that <program-id> cannot spawn other jobs with the LS boolean set.

NO-EOF
If the NO-EOF option is specified, the SMCS program does not send an end-of-file (EOF) indicatior to <program-id> after all signed-on stations have been signed off and all stations that have passed to <program-id> have logged off the SMCS program.

If a timer value is specified and no stations are signed on to <program-id> and all stations that have passed to <program-id> have logged off the SMCS program, the SMCS program waits the time period specified in the timer value before sending the EOF indicator to the program.

NO-RR

If the NO-RR option is specified, <program-id> is spawned with the RR (restrictions) attribute set to 0. Otherwise, <program-id> is spawned with the RR attribute set to 2.

If the NO-RR option is not specified, <program-id> cannot spawn jobs.

NO-SCROLL

If the NO-SCROLL option is specified, any station signing on to the program is automatically taken out of scroll mode. After the program is signed off, the station is automatically put back into scroll mode if it was in scroll mode before the sign on.

NO-ZIP

The NO-ZIP option prevents the SMCS program from executing <program-id>. If <program-id> is not currently executing when a SIGN ON, PASS, or INITIATE command is entered, then the SMCS program advises the user that the program must be executed from the system ODT. When the NO-ZIP option is specified, the <zip string> is the name of the program to which the station is to be attached.

This option is intended primarily for use with the BNA station transfer facility. Refer to the B 1000 System Burroughs Network Architecture (BNA) Installation and Operations Reference Manual for a complete description of the BNA station transfer facility.

USERCODE

The USERCODE option specifies a single usercode or partial usercode with which access to <program-id>, using the SIGN ON or PASS commands, is allowed.

USERFILE

The USERFILE option specifies a disk file containing a list of usercodes with which access to <program-id>, using the SIGN ON or PASS, commands is allowed. The disk file record size must be 90 characters; only the first 72 characters in the file are used.

If a usercode and password are required to access the file (for example, the SECURITYUSE file attribute is PRIVATE), then that usercode and password must bespecified following the USERFILE option.

<hr>

The <hr> token is the number of hours that the SMCS program is to wait before sending an EOF indicator to the program. If the <hr> token is specified, the <min> and <sec> tokens must also be specified. The <hr> token must be an integer value.

<min>

The <min> token is the number of minutes that the SMCS program is to wait before sending an EOF indicator to the program. If the <min> token is specified, the <sec> token must also be specified.
The <min> token must be an integer value between 0 and 59.

<sec>

The <sec> token is the number of seconds that the SMCS program is to wait before sending an EOF indicator to the program. The <sec> token must be an integer value.

<usercode>

The <usercode> token is the usercode permitted to sign on or pass to the program.

<partial usercode>
The <partial usercode> token is the first part of a usercode that can be specified for the USERCODE and USERFILE options. The partial usercode is terminated by a question mark (?) character. For example, if USERCODE = AB? is specified, only usercodes beginning the characters AB are allowed to sign on or pass to the program.

<file name>
The <file name> token represents the file name of the USERFILE disk file. It can be any valid B 1000 disk file identifier.

<us/pw>
The <us/pw> token represents the usercode and password required to access the USERFILE disk file.

<zip string>
The <zip string> is the string that is to be passed to the MCP by the SMCS program.

Example JOBS File:

```
$ CANDE             AUTO-START                  %
                    EXCEPT                      %
                    LOG-ON                      %
                    NO-EOF (00:10:00)           % Stay in mix for
                    NO-RR;                      %   ten minutes if no
  EXECUTE CANDE;                                %   users logged on
     FILE WORK       PID USER;                  %
     FILE RECOVERY   PID USER;                  %
     FILE AUDIT      PID USER;                  %
     FILE PAGE.FILE  PID USER;                  %
     MEMORY 5760; SW5 = 1;                      %
     PRIORITY = 13;  PROTECTED;                 %
$ SYCOM             AUTO-START                  %
                    EXCEPT                      %
                    LOG-ON                      %
                    NO-EOF                      %
                    NO-RR;                      %
  COMPILE "APTERYX" WITH SYCOM FOR SYNTAX;      %
    FILE LINK NAME LINK43;                      %
    PRIORITY = 13;  PROTECTED;                  %
$ RD                AUTO-START                  % Remote Display program
                    NO-EOF;                     %
  EXECUTE RD;                                   %
     SW=&040A00010F@;                           %
     PRIORITY = 9;                              %
$ BNA               EXCEPT                      %
                    LOG-ON                      %
                    NO-EOF                      %
                    NO-SCROLL                   %
                    NO-ZIP;                     %
  BNA/PLM;                                      %
$ PAYROLL           NO-RR                       %
                    USERCODE = WOMBAT;          % Only WOMBATs can log on
  EXECUTE PAYROLL/UPDATE                        %
  PRIORITY = 9; PROTECTED;                      %
$ UPDATE            NO-RR                       %
                    USERFILE = UPDATE/ACCESS (SUPER/STUFF);  %
  EXECUTE UPDATE/PROGRAM;                       %
  PRIORITY = 7; PROTECTED;                      %
```

The SMCS program uses <program-id> for all SIGN ON, PASS, and INITIATE commands. If a user enters ON PAYROLL, the station is attached to the program PAYROLL/UPDATE. However, if a station enters ON PAYROLL/UPDATE, the error message PAYROLL/UPDATE NOT IN THE JOBS FILE is displayed.

### Creation of the JOBS File

The JOBS file can be created using the CANDE, DMPALL, or TEXT/EDITOR programs.

## (SYSTEM)/USERCODE FILE

The list of all valid usercode/password combinations is maintained in this file. Refer to the B 1000 Systems System Software Operation Guide, Volume 2 for a detailed description of the file and how it is created.

## PROGRAM PRIORITIES

For optimum system response, it is necessary that the network controller and SMCS program run at higher priorities than other programs in the mix and that the network controller execute at a higher priority than the SMCS program. The network controller is, by default, modified by the NDL compiler to run at a priority of 15. The following example shows how to permanently assign a priority of 13 to the SMCS program.

Example:

```
MODIFY SMCS PRIORITY 13;
```

## GENERATING THE NETWORK CONTROLLER

The user must write Network Definition Language (NDL) symbolic specifications to generate the desired network controller for the SMCS program. The following guidelines are intended to aid the user in generating a proper network controller. For further information, refer to the B 1000 Systems Network Definition Language (NDL) Reference Manual.

### Network Controller Name

The network controller name is specified by the user.

### Declaration Section

The Network Information File (NIF) file name and MAX FILES can be specified in the Declaration Section. The MAX FILES declaration specifies the total number (up to a maximum of 255) of remote files that can be open at any one time. The default for the MAX FILES declaration is the number of remote files declared in the File Section of the network controller.

Example:

```
DECLARATION:
    NIF = "NIF"/"SMCS".
    MAX FILES = 20.
```

If the SYCOM program is to be used, the MAX MESSAGE SIZE declaration should be set to one-half of the SYCOM buffer size as declared in the Terminal Section.

Example:

```
MAX MESSAGE SIZE = 2421.    % SYCOM buffer size is 4842
```

## Request Section

A file of library Request procedures exists to handle common line disciplines. However, if the CANDE program is to be used in the system, the CANDE Request procedures must be declared.

Example:

```
$ LIBRARY CANDEDEC
$ LIBRARY CANDEPOLTD
$ LIBRARY CANDESELTD
```

## Control Section

A file of library Control procedures exists to handle common line-driving techniques. However, if the CANDE program is to be run in the SMCS system, the CANDE Control, CANDETDCTL, must be declared.

Example:

```
$ LIBRARY CANDETDCTL
```

## Terminal Section

The Terminal Section describes each type of remote device connected to the data communication system. The following example describes an MT 985 terminal specification.

Example:

```
TERMINAL MT985:
    TRANSMISSION = 1.
    ADDRESS      = 2.
    TYPE         = 46.
    BUFFERSIZE   = 2250.
    REQUEST      = CANDEPOLTD: RECEIVE,
                   CANDESELTD: TRANSMIT.
```

## Station Section

The Station Section defines each remote device in the system. In this section, the user defines workstations, remote ODT, and controller stations.

A workstation is defined by setting the transmit (output) and receive (input) frequency values less than 250. For example, the statement, FREQUENCY = 0, 1., where 0 is the receive frequency and 1 is the transmit frequency, defines a workstation.

A remote ODT station is defined by setting the transmit (output) frequency greater than 249 but less than 255. For example, the statement, FREQUENCY = 0, 250., where 0 is the receive frequency and 250 is the transmit frequency, defines a remote ODT station.

A controller station is defined by setting the receive (input) frequency to a value greater than 249 but less than 255. For example, the statement FREQUENCY = 250, 0., where 250 is the receive frequency and 0 is the transmit frequency, defines a controller station.

A station can be defined as both a remote ODT and a controller station by setting both the receive (input) and the transmit (output) frequency to values greater than 249 but less than 255. For example, the statement, FREQUENCY = 250, 255., where 250 is the receive frequency and 255 is the transmit frequency defines a station that is both a controller station and a remote ODT station.

These parameters are illustrated in the following examples.

The definitions for a workstation, remote ODT station, and a controller station can be incorporated into the STATION DEFAULT statement.

Example:

```
STATION DEFAULT WORKSTATION:        % workstation
   RETRY          = 25.
   FREQUENCY      = 0,0.
   MYUSE          = INPUT,OUTPUT.
STATION DEFAULT REMOTEODT:          % remcte ODT station
   RETRY          = 25.
   FREQUENCY      = 0,250.
   MYUSE          = INPUT,OUTPUT.
STATION DEFAULT CTRLRSTATION:       % controller station
   RETRY          = 25.
   FREQUENCY      = 250,0.
   MYUSE          = INPUT,OUTPUT.
STATION DEFAULT CTRLRODT:           % controller & ODT station
   RETRY          = 25.
   FREQUENCY      = 250,250.
   MYUSE          = INPUT,OUTPUT.

STATION AARDVARK:
   TERMINAL       = MT985.
   DEFAULT        = WORKSTATION.
   ADDRESS        = "T1".
STATION APTERYX:
   TERMINAL       = MT985.
   DEFAULT        = REMCTEODT.
   ADDRESS        = "T2".
STATION PLATYPUS:
   TERMINAL       = MT985.
   DEFAULT        = CTRLRSTATION.
   ADDRESS        = "T3".
STATION WOMBAT:
   TERMINAL       = MT985.
   DEFAULT        = CTRLRSPO.
   ADDRESS        = "T4".
```

## Line Section

The address of the line is specified in the Line Section along with the stations that are physically attached to that line. The address of the line and the stations that are attached to that line can be obtained from a Burroughs field engineer. If the CANDE Control Procedure, CANDETDCTL, is used, the AUTOPOLL buffer must be at least 10 times the number of stations on the line. The following is an example of a Line Section specification.

Example:

```
LINE TDLINE1:
   ADDRESS   = 02:00:00.
   CONTROL   = CANDETDCTL.
   AUTOPOLL  = 40.
   STATION   = AARDVARK,
               APTERYX,
               PLATYPUS,
               WOMBAT.
```

## File Section

The File Section defines all stations that are to be controlled by the SMCS program. The remote file name of the SMCS program is MCSREMOTE. The following is an example of the File Section.

Example:

```
FILE MCSREMOTE:
   RESIDENT = CORE.
   FAMILY   = AARDVARK,
              APTERYX,
              PLATYPUS,
              WOMBAT.
```

## Network Controller Example

The following is an example of network controller symbolic code necessary to generate a SMCS network controller.

```
DECLARATION:
     NIF =  "NIF"/"SMCS".
     MAX FILES = 20.

$ LIBRARY CANDEDEC
$ LIBRARY CANDEPOLTD
$ LIBRARY CANDESELTD
$ LIBRARY CANDETDCTL

TERMINAL MT985:
     TRANSMISSION = 1.
     ADDRESS      = 2.
     TYPE         = 46.
     BUFFERSIZE   = 2250.
     REQUEST      = CANDEPOLTD: RECEIVE,
                    CANDESELTD: TRANSMIT.

STATION DEFAULT WORKSTATION:        % workstation
    RETRY         = 25.
    FREQUENCY     = 0,0.
    MYUSE         = INPUT,OUTPUT.
STATION DEFAULT REMOTEODT:          % remote ODT station
    RETRY         = 25.
    FREQUENCY     = 0,250.
    MYUSE         = INPUT,OUTPUT.
STATION DEFAULT CTRLRSTATION:       % controller station
    RETRY         = 25.
    FREQUENCY     = 250,0.
    MYUSE         = INPUT,OUTPUT.
STATION DEFAULT CTRLRODT:           % controller & ODT station
    RETRY         = 25.
    FREQUENCY     = 250,250.
    MYUSE         = INPUT,OUTPUT.
STATION AARDVARK:
    TERMINAL      = MT985.
    DEFAULT       = WORKSTATION.
    ADDRESS       = "T1".
STATION APTERYX:
    TERMINAL      = MT985.
    DEFAULT       = REMOTEODT.
    ADDRESS       = "T2".
STATION PLATYPUS:
    TERMINAL      = MT985.
    DEFAULT       = CTRLRSTATION.
    ADDRESS       = "T3".
STATION WOMBAT:
    TERMINAL      = MT985.
    DEFAULT       = CTRLRSPO.
    ADDRESS       = "T4".

LINE TDLINE1:
    ADDRESS = 02:00:00.
    CONTROL = CANDETDCTL.
    AUTOPOLL = 40.
    STATION = AARDVARK,
              APTERYX,
              PLATYPUS,
              WOMBAT.

FILE MCSREMOTE:
    RESIDENT = CORE.
    FAMILY   = AARDVARK,
               APTERYX,
               PLATYPUS,
               WOMBAT.
```

# SETUP OF THE MCP NAME TABLE

The setup of the MCP Name Table, for use with the SMCS program, is described in the following paragraphs.

## Network Controller

For the network controller to be automatically executed by the MCP, it is necessary to specify, in the MCP Name Table, the name of the network controller. This is done by placing the name of the network controller in the C entry of the MCP name table using the following MCP command:

CM C <network controller identifier>

To verify that the C slot in the name table contains the correct name of the network controller, the following MCP command can be entered:

WW C/=

## SMCS

In order for the SMCS program to be automatically executed by the MCP after a Clear/Start operation, specify in the MCP Name Table, the name of the SMCS program and set the AMCS MCP option. The name of the SMCS program is placed into the MCS entry in the MCP name table by entering the following MCP command:

CM MCS <SMCS program name>;

To verify that the MCS slot in the name table contains the correct SMCS program identifier, the following MCP command can be entered:

WW MC/=

The AMCS MCP option is set as follows:

SO AMCS

The setting of the AMCS MCP option can be tested by entering the following:

TO AMCS

# WORKFILE NAMING CONVENTIONS

The family name of the SMCS program is automatically used as the family name for SMCS workfiles. The default names for the SMCS workfiles are listed in appendix D, Summary of Files.

For example, if the name of the SMCS program is changed to SMCS_TEST, the names of SMCS workfiles are:

| File | Workfile name |
|---|---|
| INIT.QUEUE | SMCS_TEST/INIT |
| TRACE.FILE | SMCS_TEST/TRACE |
| MCPQ | SMCS_TEST/MCPQ |
| SCHQ | SMCS_TEST/SCHQ |
| JOBQ | SMCS_TEST/JOBQ |
| MSGQ | SMCS_TEST/MSGQ |
| US.DIRECTO | SMCS_TEST/DIRECTORY |
| JOBS | SMCS_TEST/JOBS |
| GROUPS | SMCS_TEST/GROUPS |
| INFO | SMCS_TEST/INFO |
| LIST | SMCS_TEST/LIST |

This allows the user to change the name of the SMCS program without the need to modify each file individually.

# SECTION 5
# SMCS OPERATING INSTRUCTIONS

## SMCS PROGRAM EXECUTION

After the steps described in section 4, SMCS Installation, have been performed, the SMCS program can be executed from the Operator Display Terminal (ODT) using the MCP EXECUTE command.

When the SMCS program has been executed and has entered the mix, the network controller must be executing or the network controller identifier must be entered in the C slot of the MCP Name Table. If the network controller identifier is in the C slot of the Name Table, the MCP executes the network controller automatically.

The SMCS program then initializes its tables and displays on the system ODT all off-line terminals in the following format:

%SMCS = <job number> STATION <station name> OFFLINE

## SMCS PROGRAM TERMINATION

The STOP command, which is valid only from the system ODT, provides for an orderly termination of the SMCS program.

## SMCS INPUT COMMANDS

The SMCS input commands direct the SMCS program to perform various functions. The commands are presented in alphabetical order. When the syntax of the command is different for a workstation, remote ODT station, controller station, or the system ODT, separate syntax diagrams are given for each type of station and are labeled workstation syntax, remote ODT syntax, controller syntax, and system ODT syntax respectively.

In the examples following each command, the input to the SMCS program is printed in lower-case letters and is prefixed by the token I to indicate input. The output from the SMCS program is printed in upper-case letters following the output example and is prefixed by the token O to indicate output.

Table 5-1 summarizes the SMCS command set. All workstation commands are valid from a remote ODT station and a controller station. If a controller station is also a remote ODT station, all remote ODT station commands are valid from the station. All controller station commands are valid from the system ODT.

Refer to appendix H, Syntax and Notation Conventions, for an explanation of the syntax and notation conventions used in this section.

**Table 5-1. SMCS Command Summary**

| Command | Work Station | Remote ODT | Controller Station | System ODT |
|---|---|---|---|---|
| ATTACH | no | no | yes | yes |
| BROADCAST | yes | yes | yes | yes |
| BYE | yes | yes | yes | no |
| CHANGE | no | no | yes | yes |
| CLEAR | no | no | yes | yes |
| DETACH | no | no | yes | yes |
| DUMP | no | no | yes | yes |
| DYNAMIC | no | no | no | yes |
| FILE | yes | yes | yes | yes |
| HARDWARE | no | no | no | yes |
| HELP | yes | yes | yes | yes |
| ID | yes | yes | yes | no |
| INITIATE | no | no | no | yes |
| JOBS | yes | yes | yes | no |
| MACRO | yes | yes | yes | no |
| MAIL | yes | yes | yes | yes |
| MAKE | yes + | yes + | yes + | yes + |
| MOVE | no | no | yes | yes |
| NEWS | yes + | yes | yes + | yes |
| PASS | yes | yes | yes | no |
| PASSWORD | no | no | no | yes |
| READY | yes | yes | yes | yes |
| REMOVE | no | no | yes | yes |
| REPORT | yes | yes | yes | yes |
| SEND | yes | yes | yes | yes |
| SIGN ON | yes | yes | yes | no |
| SIGN OFF | yes | yes | yes | no |
| SIGNAL | no | no | yes | yes |
| STATUS | yes | yes | yes | yes |
| STOP | no | no | no | yes |
| SYSTEM | yes + | yes | yes + | yes |
| TABS | yes | yes | yes | no |
| TRACE | no | no | yes | yes |
| TRANSLATE | yes | yes | yes | no |
| USER | yes | yes | yes | no |
| ZIP | no | yes | no | no |
| ? <XX> | yes * | yes | yes * | no |
| <XX> | yes * | yes | yes * | no |

NOTES

\+ Indicates that a subset of the command is available.

* Indicates that only a subset of the MCP command set is available from that station.

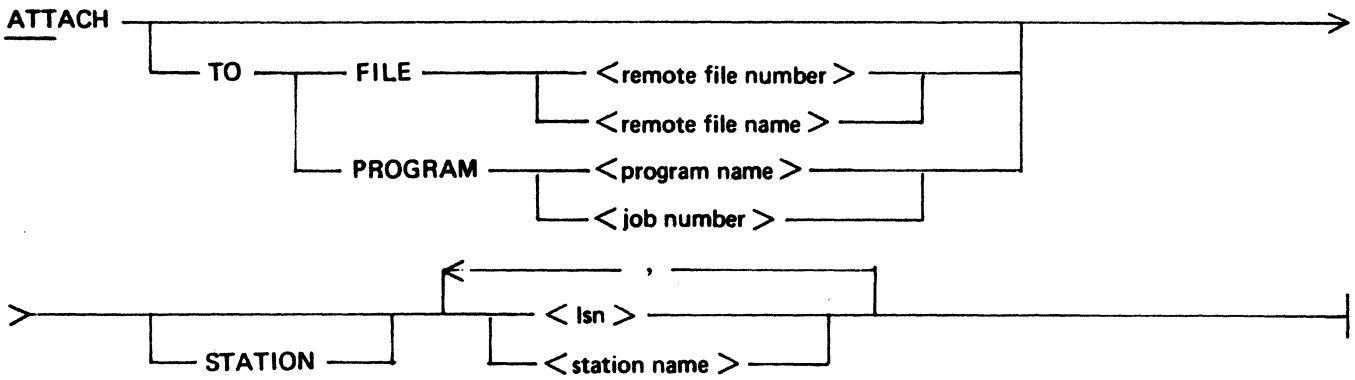
## ATTACH

The ATTACH command adds stations to a given remote file that is already open by modifying the station list of the specified remote file.

If a file or program is not specified, the station is attached to the remote file of the SMCS program.

The ATTACH command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

FILE
The FILE option attaches the station(s) to a specified remote file.

<remote file number>
The remote file number of the file to which the station(s) is to be attached.

<remote file name>
The name of the remote file to which the station(s) is to be attached.

PROGRAM
The PROGRAM option allows one or more stations to be attached to the remote file opened by the specified program.

<program name>
The name of the application program to which the station(s) is to be attached. If the program was initiated with a SIGN ON, PASS, or INITIATE command, the program name is the program-id specified in the JOBS file.

<job number>
The job number assigned to the application program to which the station is to be attached.

<lsn>
The logical station number (LSN) of the station to be attached.

<station name>
The symbolic station name of the station to be attached.

Examples:

I: attach to program text/inq 3,4
O: STATION(S) PLATYPUS, WOMBAT ATTACHED TO PROGRAM TEXT/INQ

I: attach to file remote1 station platypus, wombat
O: STATION(S) PLATYPUS, WOMBAT ATTACHED TO FILE REMOTE1

I: attach apteryx
O: STATION(S) APTERYX ATTACHED TO FILE MCSREMOTE

## BROADCAST

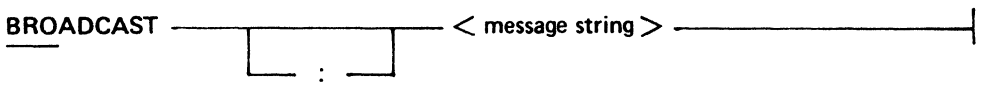
The BROADCAST command sends a message string to all stations that are attached to a remote file or that a user has logged on, using the USER or ID commands.

The SEND TO ALL command sends a message to all on-line stations including the system ODT. Refer to the discussion of the SEND command in this section for more information.

The maximum size of the message string that can be broadcast is 1920 characters, including the BROADCAST command.

The BROADCAST command is valid from all stations.

Syntax:




Semantics:

: The colon character (:) is used for documentation purposes only.

<message string>
The <message string> is the message text a user wishes to broadcast to all on-line stations. The maximum length for the BROADCAST command syntax is 1920 characters.

Examples:

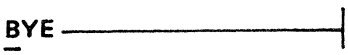
I: bro There will be a meeting in 10 minutes.
O: MESSAGE SENT TO 5 STATIONS

## BYE

The BYE command logs the current user off the SMCS program. The station must be logged on to the SMCS for this command to be valid. If the station sending the command is the last currently active station and a STOP command has previously been entered, then the SMCS program terminates after processing the BYE command.

The BYE command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Example:

```
I: BYE
O: (WOMBAT) LOGGED OFF -SMCS-AT 13:31 01/05/83
```

## CHANGE

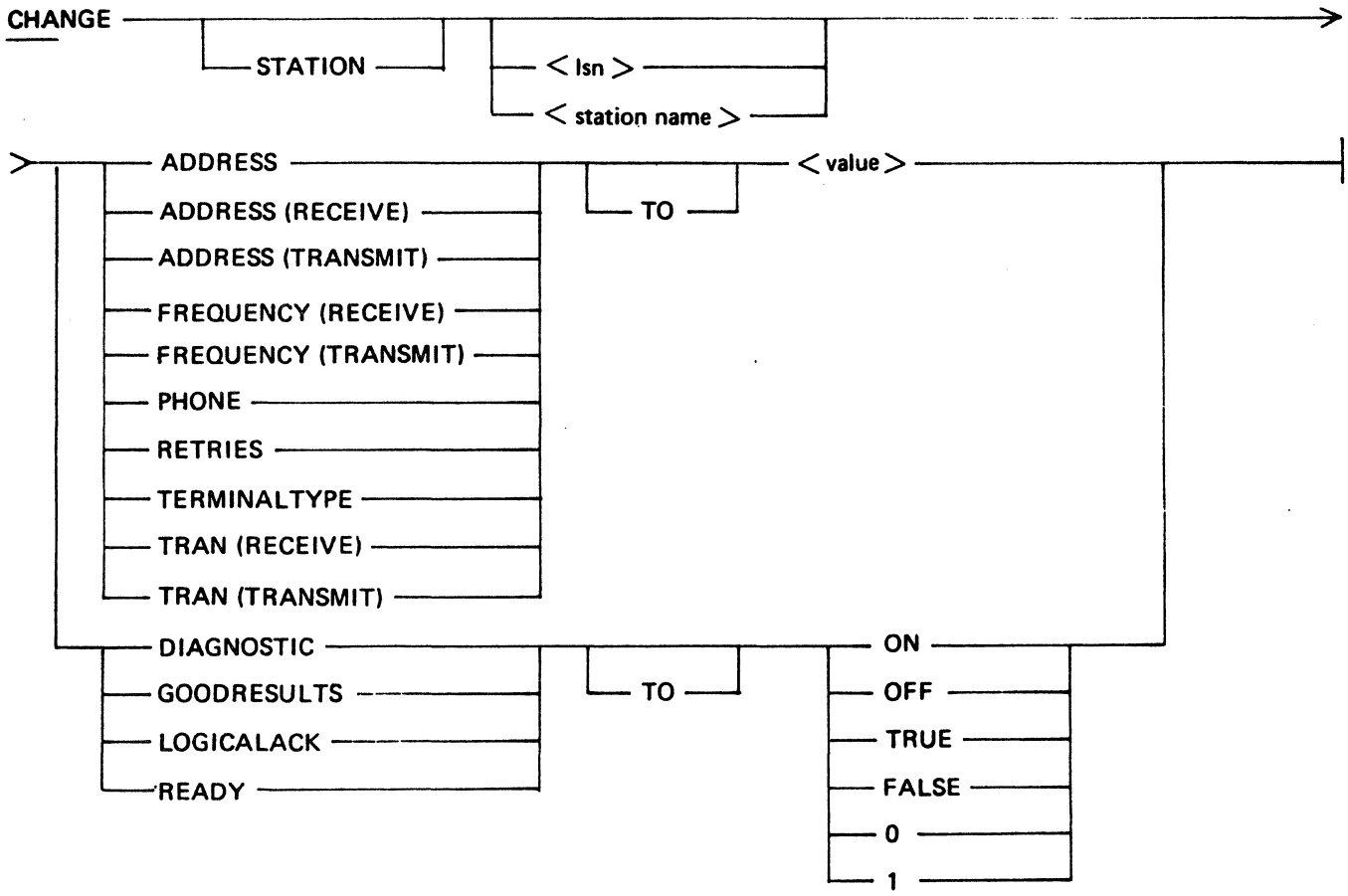
The CHANGE command dynamically changes various network controller station attributes in the network controller. These attribute changes are sent to the network controller from the SMCS by means of the MCS change message.

If the LSN and the station name are omitted, the initiating station is assumed.

Refer to the B 1000 Systems Network Definition Language (NDL) Reference Manual for more information on the various options.

The CHANGE command is valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

<lsn>
The logical station number (LSN) of the station for which the attributes are to be changed.

<station name>
The symbolic station name of the station for which the attributes are to be changed.

ADDRESS
The ADDRESS keyword specifies that the address of the specified station is to be changed. The <value> can be any address acceptable to the network controller.

ADDRESS (RECEIVE)
The ADDRESS (RECEIVE) keyword specifies that the receive address of the specified station is to be changed. The <value> can be any address acceptable to the network controller.

ADDRESS (TRANSMIT)
The ADDRESS (TRANSMIT) keyword specifies that the transmit address of the specified station is to be changed. The <value> can be any address acceptable to the network controller.

FREQUENCY (RECEIVE)
The FREQUENCY (RECEIVE) keyword specifies that the receive frequency of the specified station is to be changed. The <value> is an integer from 0 to 255.

FREQUENCY (TRANSMIT)
The FREQUENCY (TRANSMIT) keyword specifies that the transmit frequency of the specified station is to be changed. The <value> is an integer from 0 to 255.

PHONE
The PHONE keyword specifies that the telephone number of the specified station is to be changed. The <value> is a 20-character field containing the telephone number and dash (-) characters.

RETRIES
The RETRIES keyword specifies that the number of retries allowed for the specified station is to be changed. The <value> is an integer from 0 to 255.

TERMINALTYPE
The TERMINALTYPE keyword specifies that the terminal type of the specified station is to be changed. The <value> is an integer from 0 to 99. Refer to the B 1000 Network Definition Language Manual, for a complete list of the standard B 1000 terminal types.

TRAN (RECEIVE)
The TRAN (RECEIVE) keyword indicates that the receive transmission number for the specified station is to be reset to <value>. The <value> must be an integer number. The range for <value> is determined by the definition of the transmission numbers in the network controller symbolic code.

TRAN (TRANSMIT)
The TRAN (TRANSMIT) keyword indicates that the transmission number for the specified station is to be reset to <value>. The <value> must be an integer number. The range for <value> is determined by the definition of the transmission numbers in the network controller symbolic code.

DIAGNOSTIC
The DIAGNOSTIC keyword indicates that the diagnostic request procedure, specified in the network controller symbolic code, is to be enabled or disabled for the specified station.

GOODRESULTS
The GOODRESULTS keyword indicates that the goodresults facility is to be enabled or disabled for the specified station. Refer to the B 1000 Network Definition Language Manual for an explanation of the goodresults facility.

LOGICALACK

The LOGICALACK keyword indicates that the logicalack facility is to be enabled or disabled for the specified station. Refer to the B 1000 Network Definition Language Manual for an explanation of the logicalack facility.

READY

The READY keyword indicates that the specified station is to be made ready or not ready.

NOTE

The ON, TRUE, and 1 options for the DIAGNOSTIC, GOODRESULTS, LOGICALACK, and READY keywords perform the identical function of enabling the function specified by the keyword. The OFF, FALSE, and 0 options disable the function specified by the keyword.

Examples:

I: change aardvark phone 1-805-555-1212
O: STATION AARDVARK PHONE CHANGED TO 1-805-555-1212

I: change station wombat ready on
O: STATION WOMBAT READY CHANGED TO 1

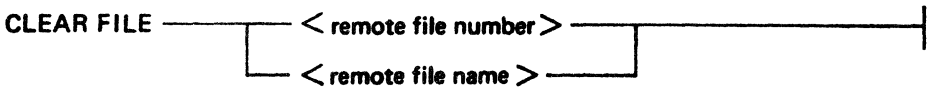
I: change station apteryx frequency(receive) to 255
O: STATION APTERYX FREQUENCY CHANGED TO 255

## CLEAR

The CLEAR command enables the user to disable, in a specified file, flags which indicate that the file is waiting for a response. When a file state is expecting an external response, such as an ATTACH_REPLY from the network controller, and the program does not respond as expected, due for example, to a program abort or other program error occurring, the CLEAR command resets the file status flag.

The CLEAR command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

<remote file number>
The remote file number of the file that is to be cleared.

<remote file name>
The name of the remote file that is to be cleared.

Examples:

I: clear file myfile
O: FILE MYFILE CLEARED

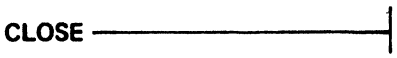
I: clear file 2
O: FILE 2 CLEARED

## CLOSE Command

The CLOSE command closes the JOBS files so that a user can make changes to the JOBS file.

The JOBS command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Example:

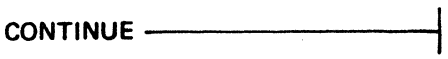
```
I: close
```

## CONTINUE Command

The CONTINUE command continues a step Macro procedure that was suspended by a WAIT CONTINUE step macro command. Refer to section 6, Macro Procedures, for a discussion of macro procedures.

The CONTINUE command is valid from all stations except the system ODT.

Workstation, Remote ODT, and Controller syntax:




Example:

```
I: continue
```

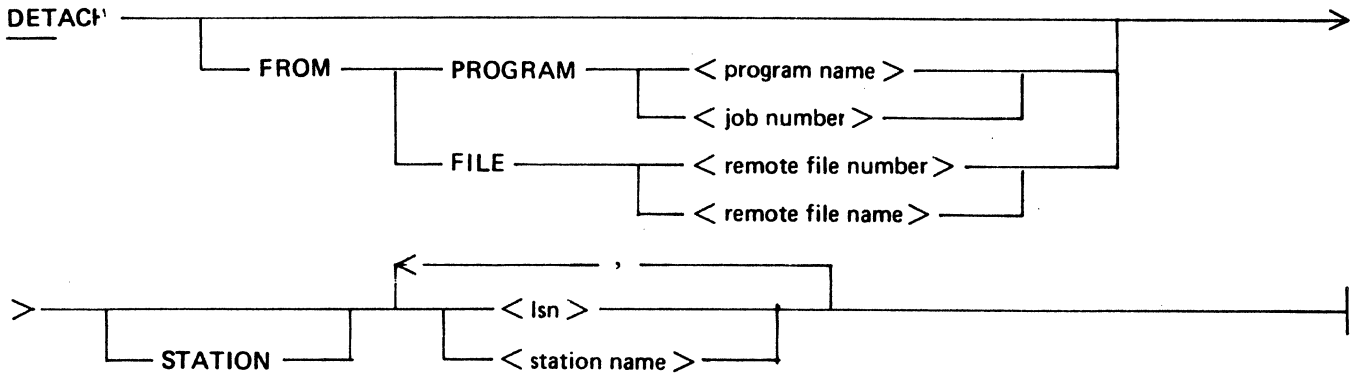
## DETACH

The DETACH command removes a station from a specified remote file or application program. The SMCS program does this by modifying the specified remote station file list.

If a program or file is not specified, the station(s) is detached from the remote file of the SMCS program.

The DETACH command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

PROGRAM
The PROGRAM option allows a station(s) to be detached from the remote file opened by the program.

<program name>
The <program name> is the name of the program from which the station is to be detached. If the program was initiated with a SIGN ON, PASS, or INITIATE command, the program name is the program-id specified in the JOBS file.

<job number>
The <job number> is the MCP-assigned job number of the program from which the station is to be detached.

FILE
The FILE option detaches the station(s) from a specified remote file.

<remote file number>
The <remote file number> is the remote file number of the file from which the station is to be detached.

<remote file name>
The <remote file name> is the name of the remote file from which the station is to be detached.

<lsn>
The <lsn> is the logical station number of the station that is to be detached.

<station name>
The <station name> is the symbolic station name of the station that is to be detached.

Examples:

I: detach from program cande platypus
O: STATION(S) PLATYPUS DETACHED FROM PROGRAM CANDE

I: detach from file candequeue wombat
O: STATION(S) WOMBAT DETACHED FROM FILE CANDEQUEUE

I: detach from file rem1 station apteryx, wombat
O: STATION(S) APTERYX, WOMBAT DETACHED FROM FILE REM1

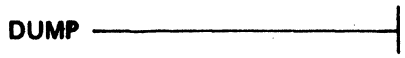
I: detach aardvark
O: STATION(S) AARDVARK DETACHED FROM FILE MCSREMOTE

## DUMP

The DUMP command initiates an SMCS program memory dump.

The DUMP command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Example:

```
I: dump
O: SMCS DUMP FILE CREATED: DUMPFILE/1234
```

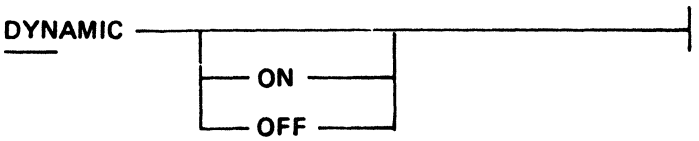
## DYNAMIC

If a station encounters an exception, the DYNAMIC command informs the SMCS program whether the station should be made not ready by the network controller.

If no option is entered, the current status of the DYNAMIC option is displayed. The default for this option is ON.

The DYNAMIC command is valid only from the system ODT.

System ODT syntax:




Semantics:

ON
The ON option causes a station that encounters an exception to be made not ready by the network controller.

OFF
The OFF option causes a station that encounters an exception, such as consecutive timeouts, to be automatically made not ready by the SMCS program. The SMCS program displays a message on the system ODT indicating that the station is off line.

Examples:

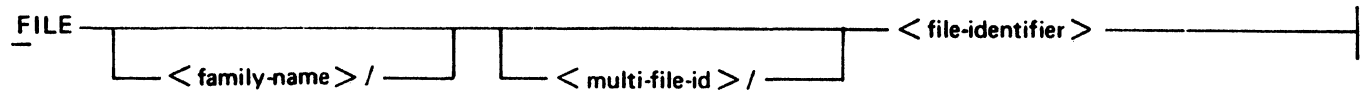
```
I: dynamic
O: DYNAMIC = ON

I: dynamic on
O: DYNAMIC = ON

I: dyn off
O: DYNAMIC = OFF
```

## FILE

The FILE command displays current information for a specified file.

The FILE command is valid from all stations.

Syntax:




Semantics:

< family-name >
The user disk identifier of the specific disk on which the file is located. The < dp-id > is blank or omitted for files on the system disk.

< multi-file-id >
An identifier that is the name given to identify a main directory file name with subdirectory entries.

< file-identifier >
A subdirectory file name in the disk file directory or, if it appears alone, the main directory file name.

Example input:

```
 I: file smcs/jobs
O: 'SMCS/JOBS' IS A PRIVATE FILE OF
    229 RECORDS (90 BYTES, 20 PER BLOCK) IN
    12 BLOCKS (12 PER AREA, 1 AREA) CREATED ON
    12/14/82 , LAST ACCESSED ON
    01/05/83 (ACCESS SECURITY IS INPUT-OUTPUT) AND CURRENTLY
    IN USE BY 2 USERS
```

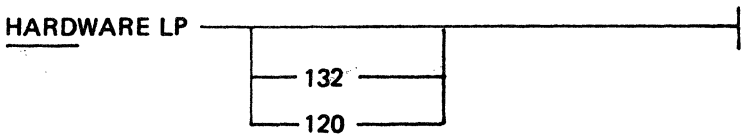
## HARDWARE

The HARDWARE command specifies the record size of the disk or printer TRACE file. The HARDWARE command controls the utilization of the system line printers by the SMCS program in order to take advantage of a 132 print position line printer if it is available.

The default for the HARDWARE command, which is kept in the INFO file, is 120 characters.

The HARDWARE command is valid only from the system ODT.

System ODT syntax:




Examples:

I: hardware lp 132
O: LINE PRINTER ATTRIBUTE CHANGED TO 132

I: hard lp
O: LINE PRINTER WIDTH IS 132

## HELP

The HELP command causes a list of all valid commands for that station to be displayed.

The HELP command is valid from all stations.

Syntax:

Example:

```
I: help
```

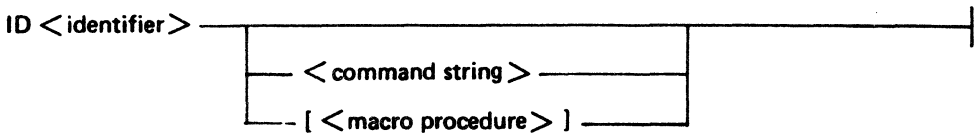
## ID

The ID command establishes a station identifier. A command string can be entered and is treated as a normal input message to the SMCS after the ID command is processed. If the identifier is already active at another station, the user is notified that the identifier is active at another station.

The SMCS program does not scan for macro procedures delimited by the bracket characters ([ ]) in the ID command. This means that the macro procedure invocation is not expanded before the ID command is processed and, so, allows a user to log on with a macro procedure following the ID command and have it processed after the log-on is complete.

The ID command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

<identifier>
The <identifier> is a unique identifying name given to the station by the user.

<command string>
The <command string> is any allowable SMCS command including associated keywords and variable data.

<macro procedure>
The <macro procedure> is any valid macro procedure for the usercode of the station. Refer to section 6, Macro Procedures, for more information on macro procedures.

Examples:

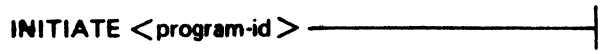
```
I: id apteryx
I: id apteryx help
I: id apteryx [on]
O: B1000 SMCS MARK 10.0.007 (08/26/82 11:23) SYSTEM-ID: NIRVAN
    "APTERYX" LOGGED ON -SMCS-AT 14:04 01/05/83
    YOU ARE AT STATION 11 (TD832)
    NEWS: There is a meeting at 1500 this afternoon.

    <command output messages>
```

## INITIATE

The INITIATE command initiates a job from the system ODT. The specified program must be declared in the JOBS file and is spawned by the SMCS program in the same way a job is automatically spawned by the SMCS program as a result of the AUTO-START option being specified in the JOBS file for that job. No stations are initially attached to the remote file of the initiated program and any station can thereafter either PASS or SIGN ON to the program.

The INITIATE command is valid only from the system ODT.

System ODT syntax:




Semantics:

<program-id>
This field is a program name that is declared in the JOBS file.

Example:

I: initiate cande
O: PROGRAM SPECIFIED HAS BEEN INITIATED

## JOBS

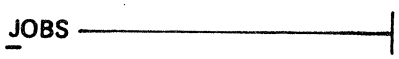
The JOBS command returns information about all jobs in the system, both active and scheduled.

The JOBS command allows a user to observe the total activity on the local system, not limited by the security imposed by usercode. Since this is a passive command (unsecured MCP communication), it does not affect the local system activity.

When the JOBS input command is processed by the SMCS program, the SMCS first determines if the RD program is executing and, if so, if it can support another passed station. If these conditions are true, then the request is passed to the RD program for a response; otherwise, the SMCS program requests the function of the MCP.

The JOBS command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Example:

```
I: jobs
O: GEM/NC_5JAN  =2816 PP=15, MP=15 "WAIT" STATUS
    SYSTEM/ODT  =2817 PP=14, MP=14 WAITING FOR I/O COMPLETE
    SMCS  =2818 PP=13, MP=13 EXECUTING
    (GEM) *CANDE2  =2822 PP=11, MP=11 "WAIT" STATUS
    SYCOM : "PA&S"  =2820 PP=9, MP=9 "WAIT" STATUS
```

## LOGON Command

The LOGON command adds, changes, or deletes the default log-on command. The default log-on command can be a macro procedure.

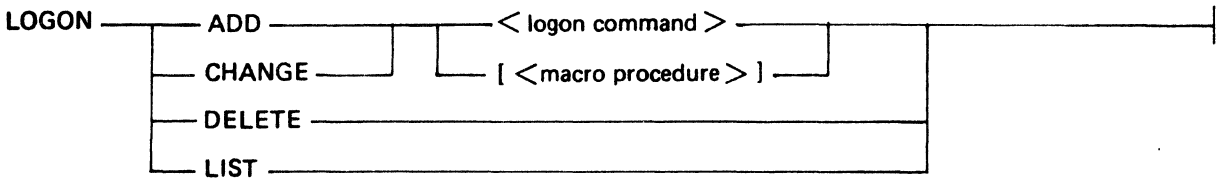
The SMCS program can automatically perform a predefined SMCS command when a user logs on to the SMCS program. This command is the default log-on command.

If an SMCS command is entered following the USER command, the SMCS command is processed before the default log-on command is processed.

In order to use the default log-on command, program switch 3 must be set to 1 at beginning of job or the MACRO ON command specified prior to the log-on of the user.

The default log-on command is restricted to 90 characters. The SMCS program does not scan for the bracket characters ([ ]) in the LOGON command, so a macro procedure can be used as the default log-on command.

Workstation, remote ODT, and controller syntax:




Semantics:

ADD
The ADD keyword causes the log-on command specified by the <logon command> token to be added to the DIRECTORY file.

CHANGE
The CHANGE keyword causes the default log-on command to be changed to the log-on command specified by <logon command>.

DELETE
The DELETE keyword causes the default log-on command to be deleted.

LIST
The LIST keyword causes the default log-on command to be displayed on the station.

<logon command>
The <logon command> token is the default log-on command to be added or changed.

<macro procedure>
This field can be any valid macro procedure for the usercode of the station. Refer to section 6, Macro Procedures, for more information on macro procedures.

Examples:

I: log delete
O: LOG-ON DEFAULT COMMAND DELETED

I: logon add tabs 10+10
O: LOG-ON DEFAULT COMMAND ADDED

I: logon change tabs 10+5
O: LOG-ON DEFAULT COMMAND CHANGED

I: logon add [lo]
O: LOG-ON DEFAULT COMMAND ADDED

I: logon list
O: LOG-ON DEFAULT COMMAND: [LO]

## MACRO

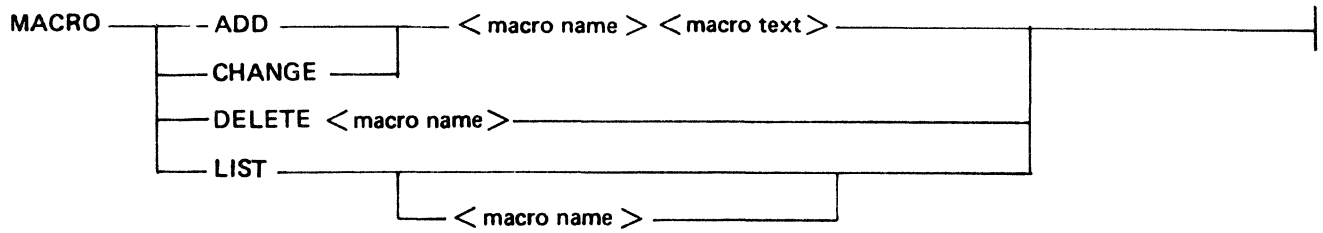
The MACRO command adds, changes, or lists macro procedures stored in the DIRECTORY file. The MACRO command is valid from all station types. Macro procedures are stored under the usercode used to log on and can only be referenced by control strings that are zipped with that usercode. A macro procedure can also be referenced by an ID identical to the usercode under which the macro procedure was created.

The ON keyword enables the macro procedure function if program switch 3 was reset at beginning of job. This keyword is only valid at a controller station or the system ODT.

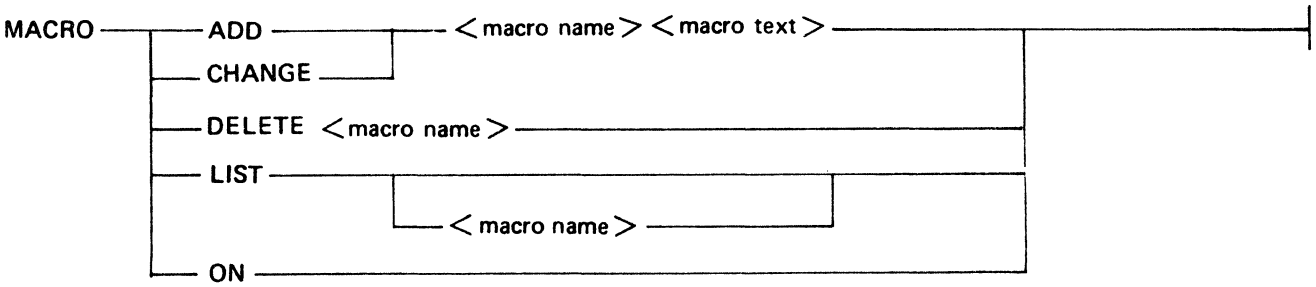
If column 80 in a<macro text> record is a nonblank character when the ADD or CHANGE keywords are specified, a step in a step macro procedure is indicated.

Refer to section 6, Macro Procedures, for more information on the SMCS macro and step macro functions.

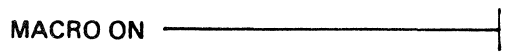
Workstation and remote ODT syntax:




Controller syntax:




System ODT syntax:




Semantics:

ADD

The ADD option writes <macro text> into a portion of the DIRECTORY file under the usercode of the station and <macro name>.

CHANGE

The CHANGE option changes an existing macro procedure, <macro name>, to <macro text>. The macro procedure to be changed is deleted and rewritten in the DIRECTORY file.

DELETE

The DELETE option deletes an existing macro procedure, <macro name>, from the DIRECTORY file.

LIST

The LIST option displays a directory listing of all existing macro procedures for the usercode of the station or lists the contents of a specific macro procedure. If <macro name> is not specified, a directory list of all macro procedures for the usercode of the station is displayed with the date and time that the macro procedure was created.

ON

The ON keyword enables the macro procedure and default log-on functions if program switch 3 was reset at beginning of job. This keyword is only valid at a controller station or the system ODT.

<macro name>

<macro name> is the token that references the macro procedure. It must be ten characters or less and must begin with an alphabetic character.

<macro text>

<macro text> is the macro procedure that is invoked when the <macro name> is referenced. <macro text> can be up to 23 lines of text and must be 1000 characters or less. The presence of a nonblank character in column 80 indicates to the SMCS program that the line <macro text> is part of a step macro procedure.

Examples:

```
I: macro add cobol
      compile p cobol library;
        file cards disk name p-s;
        file line u.n no hardware no atp name #p;
O: MACRO ADDED

I: macro change cobol
      compile p with cobol library;
        file cards disk name c;
        file source name s;
        file line u.n no hardware no atp name #p;
O: MACRO CHANGED

I: macro list
O: MEMO 01/05/83 10:02:11.4
   TELEX 01/05/83 10:02:29.1
   REPLY 01/05/83 10:02:34.0
   NFS 01/05/83 10:02:39.7
   LETTER 01/05/83 10:02:56.8

I: macro list cobol
O: COMPILE P WITH COBOL;
   FILE CARDS DISK NAME C;
   FILE SOURCE NAME S;
   FILE LINE U.N NO HARDWARE NO ATP NAME #P;

I: macro delete cobol
O: MACRO DELETED

I: macro on
O: MACRO OPTION ON
```

## MAIL

The MAIL command writes messages to the MAIL file.

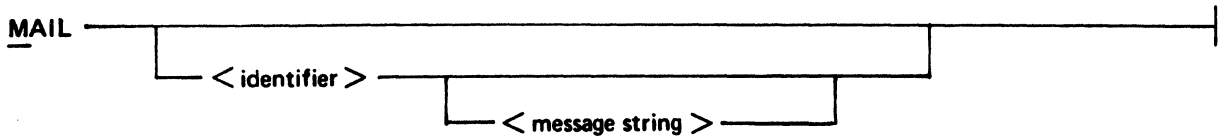
If an identifier is not specified in the command string, the user station identifier is assumed.

If both a message string and an identifier are not included in the command string, the current mail count for the station is displayed.

A complete description of the MAIL file is presented in appendix E of this document.

The MAIL command is valid from all stations.

Syntax:




Semantics:

<identifier>
A unique identifying name given to a station or a valid usercode.

<message string>
Message text to be mailed to a specific identifier.

Examples:

I: mail
O: USER WOMBAT CURRENT MAIL COUNT = 4

I: mail aardvark this is a mail message
O: MESSAGE MAILED TO AARDVARK

## MAKE

The MAKE command changes certain station attributes. The attribute to be changed is specified with the command.

A workstation and a remote ODT station can change the signal character and enable or disable reverse scrolling.

A controller station can change the signal character, enable or disable reverse scrolling, make a station ready or not ready, and enable or disable a station as a controller station.

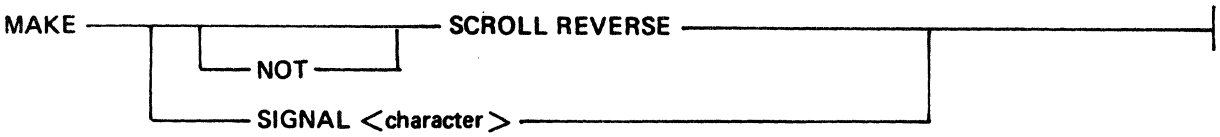
The system ODT can make a station ready or not ready and enable or disable a station as a controller or a remote ODT station.

The signal character of the station is used for communications with the SMCS program while the station is signed on to a program.

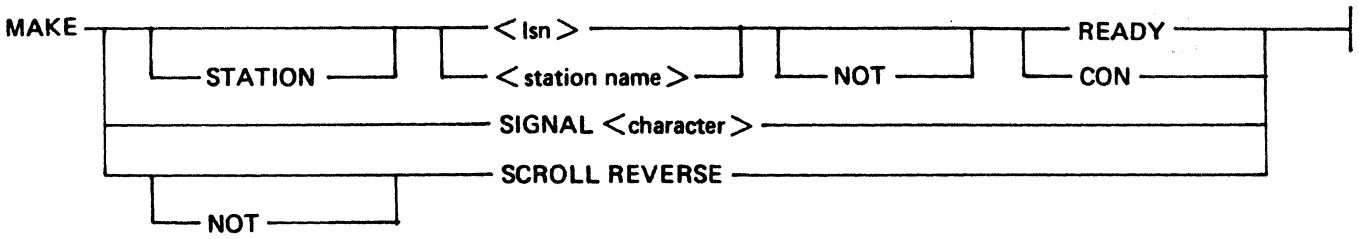
To change reverse scrolling, the initiating station cannot be in forward scroll mode when the MAKE command is entered.

The signal character for the station can only be changed when the station is signed on to a program, and the change is only effective while the station is attached to the program. The signal character is initially set to an asterisk (*) character for all stations.

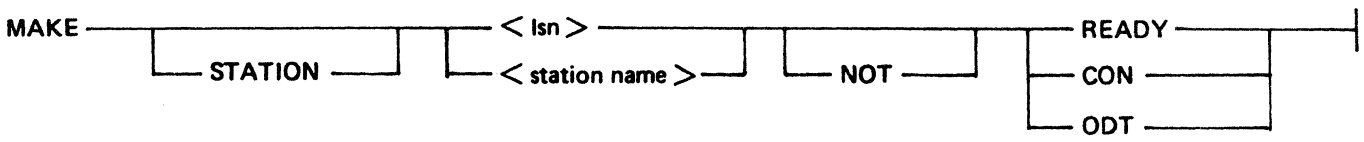
Workstation and remote ODT syntax:




Controller syntax:




System ODT syntax:



Semantics:

NOT
The NOT keyword indicates that an option is to be disabled.

REVERSE SCROLL
The REVERSE SCROLL option indicates that reverse scrolling is to be enabled or disabled.

SIGNAL
The SIGNAL keyword specifies that the signal character is to be changed.

<character>
The <character> is any character which is acceptable for a signal character except a blank or a period (.) character. Refer to appendix B for a list of allowable signal characters.

READY
The READY option is used by a controller station or the system ODT to make a station ready or not ready.

CON
The CON option is used by a controller station or the system ODT to make a station a controller station, or to make a controller station not a controller station.

ODT
The ODT option is used by the system ODT to make a station a remote ODT station, or to make a remote ODT station not a remote ODT station.

<lsn>
The <lsn> is the logical station number (LSN) of the station for which the MAKE command is to be performed.

<station name>
The <station name> is the symbolic station name of the station for which the MAKE command is to be performed.

Examples:

```
I: make scroll reverse
O: REVERSE SCROLL ENABLED

I: make not scroll reverse
O: REVERSE SCROLL DISABLED

I: make signal $
O: STATION PLATYPUS SIGNAL CHANGED TO @

I: make aardvark ready
O: STATION AARDVARK READY CHANGED TO 1

I: make aardvark not ready
O: STATION AARDVARK READY CHANGED TO 0

I: make platypus con
O: STATION PLATYPUS FREQUENCY(RECEIVE) CHANGED TO 250
```

I: make platypus not con
O: STATION PLAYTYPUS GEM FREQUENCY(RECEIVE) CHANGED TO 249

I: make apteryx odt
O: STATION APTERYX FREQUENCY(TRANSMIT) CHANGED TO 250

I: make apteryx not odt
O: STATION APTERYX FREQUENCY(TRANSMIT) CHANGED TO 249

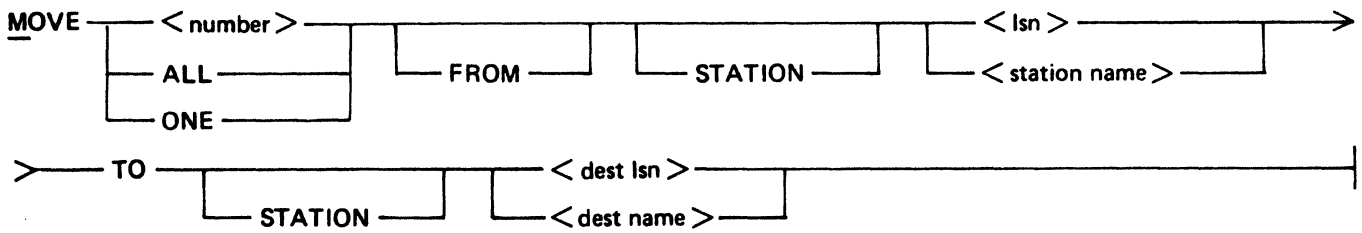
## MOVE

The MOVE command enables a station to reroute output messages and requeue all or part of the output of one station into the output queue of another station.

An integer number (or ALL) can be specified representing the number of messages to be moved which are the oldest messages in the output queue.

The MOVE command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

<number>
The <number> is an integer representing the number of messages to be moved to the specified station.

ONE
The ONE option requeues the oldest message in the output queue.

ALL
The ALL option requeues all the output of one station into the output queue of another station.

<lsn>
The <lsn> is the logical station number (LSN) of the station from which the messages are to be moved.

<station name>
The <station name> is the symbolic station name of the station from which the messages are to be moved.

<dest lsn>
The <dest lsn> is the logical station number of the station to which the messages are to be moved.

<dest name>
The <dest name> is the symbolic station name of the station to which the messages are to be moved.

Examples:

I: move all from wombat to aardvark
O: 3 MESSAGE(S) MOVED FROM WOMBAT TO AARDVARK

I: move one from apteryx to platypus
O: 1 MESSAGE(S) MOVED FROM APTERYX TO PLATYPUS

I: move all from station 6 to station 7
o: 16 MESSAGE(S) MOVED FROM APTERYX TO WOMBAT

## NEWS

The NEWS command displays or changes the contents of the current system news message.

A workstation or a controller station can only display the current system news message.

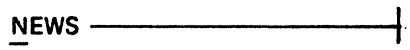
A remote ODT station or the system ODT can display, clear, or change the current system news message.

The system news message is automatically displayed when a user logs on.

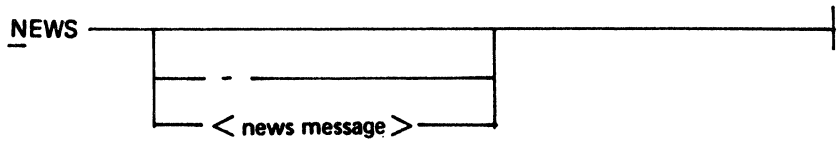
The maximum size of the system news message is 140 characters and the system news message is stored in the INFO file.

If no option is specified in the command string, the current system news message is displayed.

Workstation and controller syntax:




Remote ODT and system ODT syntax:




Semantics:

<news message>
A news message to be displayed when a station logs on to the SMCS program.

minus sign (-)
Used to clear the news message area in memory.

Examples:

I: news
O: NEWS: THE NEW AP300 IS NOW AVAILABLE FOR USE.

I: news -
O: NEWS MESSAGE CLEARED

I: news Good morning, there will be a meeting a 10:00 am.
O: NEWS: GOOD MORNING, THERE WILL BE A MEETING A 10:00 AM.

## PASS

The PASS command requests a function of a program in the local system without signing on to that program.

The program identifier specified must be declared in the JOBS file. If the program is not executing, then the SMCS uses the zip string specified in the JOBS file for the program identifier and makes the appropriate entry in the program table. If the program is already executing and the value of the MAX.STATIONS file attribute has been reached, a new copy of the program is spawned by the SMCS program. Otherwise, the copy of the program currently executing is used.

A maximum of 1920 characters minus the length of the pass string can be passed to a program.

When a BYE command is entered by a station that is logged on to the SMCS program and that has entered the PASS command, the station is removed from the list of known passers for each program to which it had passed. For any such programs, a remote file station count of zero indicates that there are no stations signed on and no stations that have passed to the program are still logged on to the SMCS. When the remote file station count goes to zero, the SMCS sends an EOF indicator to the program unless the NO-EOF option is set in the JOBS file for that program. When a user signs on to a program, the station is attached to the same copy of the program to which the user had been passing using the PASS command, if such a copy exists.

The PASS command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:

Semantics:

<program-id>
The program identifier as specified in the JOBS file.

<data string>
Message data text to be passed. The maximum length for the PASS command syntax is 1920 characters including the data string. The <data string> can be a blank or null character string.

Examples:

```
I: pass rd omx

I: pass sycom .what
```

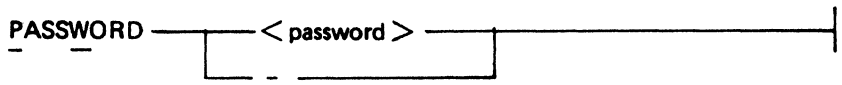
## PASSWORD

The PASSWORD command causes a password to be set for COBOL74 data communcations users.

The password set with this command is used when the SMCS program receives an Open request for a COBOL74 program. For all COBOL74 ENABLE__DISABLE requests forwarded to the SMCS from the network controller that involve the password option, the SMCS program examines the password field which it maintains in the INFO file. If this master password is blank, no check is made; otherwise, password validation is performed. If the password in the Open request from the network controller is not the same as the password for the SMCS program, the SMCS program denies the Open request. The user must recompile the COBOL74 program using the password specified by the PASSWORD command or change the password using the PASSWORD command to match the password expected by the COBOL74 program.

The PASSWORD command is only valid from the system ODT.

System ODT syntax:




Semantics:

< password >
This field is a data string limited to ten characters.

-

The hyphen character clears the COBOL74 password.

Examples:

I: password cobol74
O: COBOL '74 PASSWORD: COBOL74

I: password turkey
O: COBOL '74 PASSWORD: TURKEY

I: password -
O: COBOL '74 PASSWORD CLEARED

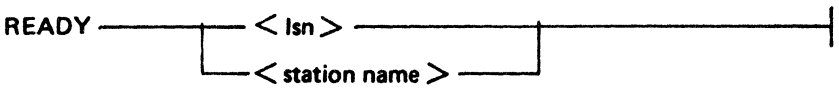
## READY

The READY command causes a change request to be sent to the network controller for the logical station number (LSN) or symbolic station name to make the station ready.

Initially, all stations are assumed on line and ready. If a station becomes unresponsive to the network controller, the SMCS is notified and the message STATION <lsn> IS OFFLINE is displayed on the system ODT. The station is also marked off line by the network controller. The entry of this command for the appropriate station from another station causes the network controller to attempt to make the station ready.

The READY command is valid from all stations.

Syntax:




Semantics:

<lsn>
The logical station number of the station that is to be made ready.

<station name>
The symbolic station number of the station that is to be made ready.

Examples:

I: ready apteryx
O: STATION APTERYX READY CHANGED TO 1

I: ready 69
O: STATION WOMBAT READY CHANGED TO 1

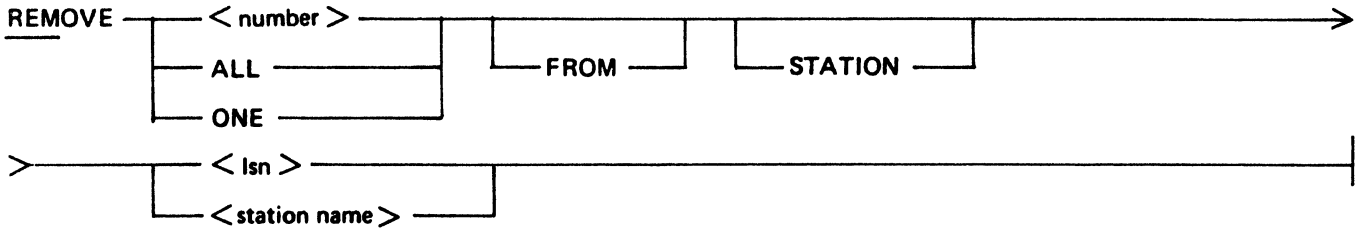
## REMOVE

The REMOVE command purges all or part of the output queue of a given station.

An integer number or ALL can be specified, representing the number of messages or all of the messages. The messages purged are the oldest messages in the output queue.

The REMOVE command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

<number>
The <number> is an integer representing the number of messages to be removed.

ALL
The ALL option purges all of the output queue of a station.

ONE
The ONE option purges the oldest message from the output queue of the station.

<lsn>
The <lsn> is the logical station number (LSN) of the station from which the messages are to be removed.

<station name>
The <station name> is the symbolic station name of the station from which the messages are to be removed.

Examples:

I: remove all from aardvark
O: 4 MESSAGE(S) REMOVED FROM AARDVARK

I: rem one from station wombat
O: 1 MESSAGE(S) REMOVED FROM WOMBAT

I: remove 6 from station apteryx
O: 6 MESSAGE(S) REMOVED FROM APTERYX

## REPORT

The REPORT command displays information in various configurations about the stations under the control of the SMCS program.

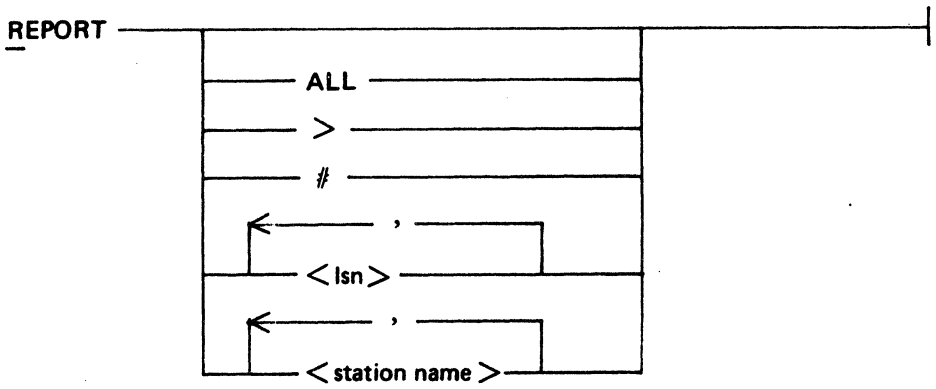
When the REPORT command is entered from a workstation, the station name, station usercode or identifier, and the program to which the station is signed on is displayed.

When the REPORT command is entered from a remote ODT station, a controller station, or the system ODT, the station type is displayed in addition to the station name, station usercode or identifier, and the program to which the station is signed on. The station type is enclosed in parentheses and indicates whether the station is a remote ODT (ODT), a controller station (CON), or both a controller station and a remote ODT (C&O).

If no option is used in the command, a list of current on-line stations is displayed.

The REPORT command is valid from all stations.

Syntax:




Semantics:

ALL
The ALL option causes information on all stations controlled by the SMCS program to be displayed.

>
The > option causes information on the station from which the REPORT command was entered to be displayed.

#
The # option causes the total number of stations controlled by the SMCS program to be displayed.

<lsn>
The <lsn> option causes information on the stations specified by the logical station number (LSN) to be displayed.

<station name>
The station name option causes information on the stations specified by the symbolic station name to be displayed.

Examples:

```
I: r
O: AARDVARK (ODT): (THX1138) ON JOB #7576 CANDE
   APTERYX (CON): (APTERYX) <ONLINE>
   PLATYPUS : (WOMBAT) <ONLINE>
 >WOMBAT (C&O): (WOMBAT) <ONLINE>
 4 STATIONS REPORTED AT 08:10 01/06/83 SYSTEM-ID: NIRVANA
I: report >
O: >WOMBAT (C&O): (WOMBAT) <ONLINE>
   1 STATION REPORTED AT 08:10 01/06/83 SYSTEM-ID: NIRVANA
I: report #
O: 4 STATIONS REPORTED AT 08:10 01/06/83 SYSTEM-ID: NIRVANA
I: report all
O: AARDVARK (ODT): (THX1138) ON JOB #7576 CANDE
   APTERYX (CON): (APTERYX) <ONLINE>
   PLATYPUS : (WOMBAT) <ONLINE>
 >WOMBAT (C&O): (WOMBAT) <ONLINE>
   DATEENTRY : <ONLINE>
   VSIN1 (C&O): <OFFLINE, READY>
   VSIN2 (C&O): <OFFLINE, READY>
 7 STATIONS REPORTED AT 08:10 01/06/83 SYSTEM-ID: NIRVANA
I: report aardvark, apteryx
O: AARDVARK (ODT): (THX1138) ON JOB #7576 CANDE
   APTERYX (CON): (APTERYX) <ONLINE>
 2 STATIONS REPORTED AT 08:10 01/06/83 SYSTEM-ID: NIRVANA
```

## SEND

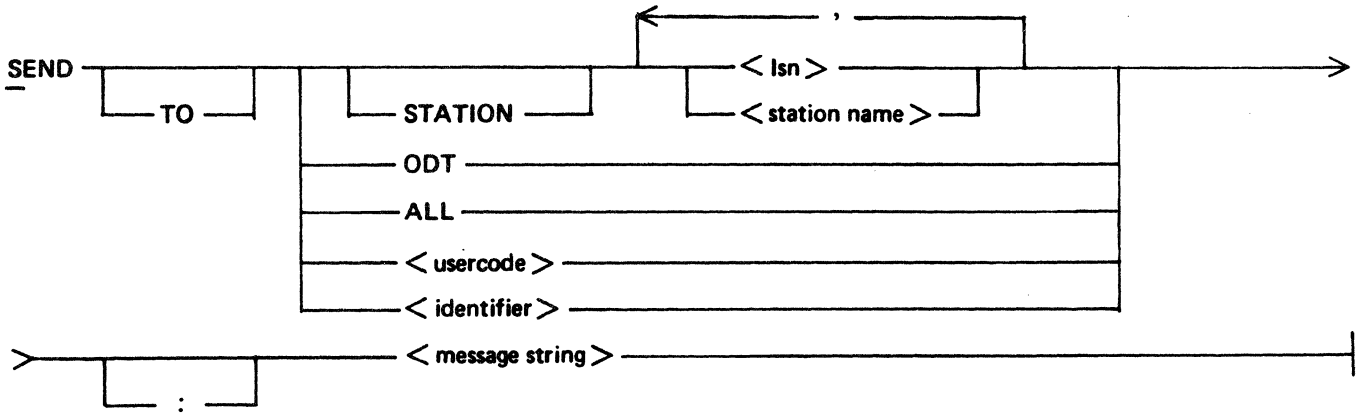
The SEND command writes messages to stations that are assigned to the SMCS program in the network controller or that have been attached to the SMCS program. A message can be sent to all stations in the network or to selected ones. The stations are specified by the station name or logical station number (LSN) assigned in the network controller.

The SMCS program queues messages for the selected stations and sends an acknowledgment to the originating station. The acknowledgment does not imply that all stations received the message. It only signifies that the message was queued for output for the specified stations.

The maximum message length that can be sent is 1920 characters including the SEND command and the station list.

This command is valid from all stations.

Syntax:




Semantics:

<lsn>
The <lsn> is the logical station number (LSN) of the station to which the message is to be sent.

<station name>
The <station name> is the symbolic station name of the station to which the message is to be sent.

ODT
The ODT option causes the message to be sent to the system ODT.

ALL
The ALL option causes the message to be sent to all stations including the system ODT.

<usercode>
The <usercode> is the usercode of the station to which the message is to be sent. The usercode is the usercode which was entered with the USER command.

<identifier>
The <identifier> is the identifier of the station to which the message is to be sent. The identifier is the identifier that was entered with the ID command.

:

The colon character (:) is used for documentation purposes only.

<message string>

The message string is the message that is to be sent to another station or stations. The maximum length of the message string is 1920 characters including the SEND command and the station list.

Examples:

I: send aardvark what is an aardvark?
O: MESSAGE SENT TO 1 STATION

I: send to odt please mount my tape
O: MESSAGE SENT TO 1 STATION

I: send to all datacomm is now avaliable
O: MESSAGE SENT TO 99 STATIONS

I: send to (myuser) are you there?
O: MESSAGE SENT TO 1 STATION

## SIGN ON

The SIGN ON command attaches a station to an application program. The program identifier specified must be declared in the JOBS file. A station can be attached to only one program at a time. However, a program can have several stations attached at the same time.

When the SMCS program receives the SIGN ON command, it goes through its tables to determine if the program specified has been initiated. If the program is currently executing and has opened a remote file, the station is attached to that remote file. This is accomplished by an attach message sent to the network controller by the SMCS if the remote file is capable of handling multiple stations; otherwise, another copy of the program is spawned by the SMCS program. The station then receives acknowledgement, in the form of a reply message, indicating that it is signed on.

If the program has not been initiated, the SMCS program uses the zip string specified in the JOBS file for the program identifier through an MCP ZIP command and makes the appropriate entry in the program table. When the program comes into the mix and opens its remote file, the remote file Open request is received by the SMCS program. The SMCS program then attaches all stations that are waiting to sign on to this program by approving the remote file Open request for these stations. The station then receives an acknowledgement message indicating that it is signed on to the program.

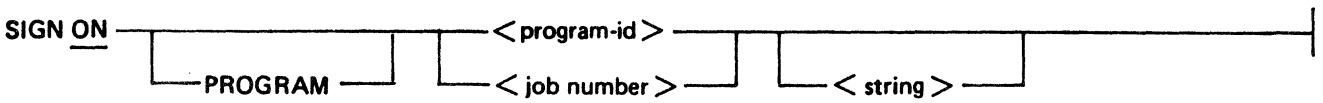
If the program has been initiated but the remote file open request has not been received by the SMCS program, then the SMCS program links this station to the other stations waiting for the remote file to be opened by the program.

All messages entered at a signed-on terminal, except those messages beginning with the SMCS signal character, are sent directly to the application program. Those messages which begin with the SMCS signal character are sent to the SMCS program for processing. All reply messages from the application program are sent directly to the station of origin.

If <string> is specified in the command syntax and the LOG-ON option is specified for the program in the JOBS file, then the <string> is sent as second input to the program; otherwise, the <string> is sent as first input to the program.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

<program-id>
The program identifier as specified in the JOBS file.

<job number>
The job number assigned to the program by the MCP.

<string>
Data such as usercode/password that is to be sent to the program to which the station is to be signed on.

Examples:

```
I: sign on cande get myfile
O: B1000 CANDE MARK 10.0.005 (11/04/82 10:40)
   USER WOMBAT LOGGED ON AT 08:25:04.2 AM 01/06/83.
   YOU ARE AT STATION 54 (TD832)
   FILE MYFILE LOADED
```

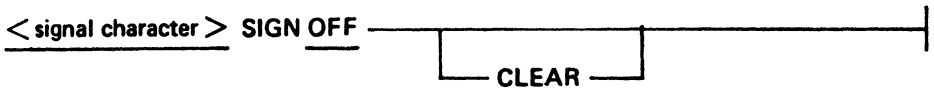
## SIGN OFF

The SIGN OFF command detaches the station from an application program. The SIGN OFF command is valid from all stations except the system ODT. When detachment is complete, the station is sent an acknowledgment that the sign-off is complete. Unless the NO-EOF option is set in the JOBS file, the application program is sent an EOF message by the SMCS program whenever the station count is zero, which indicates that there are currently no active stations signed on to it and no stations which have passed to the program are still logged on to the SMCS program. Since the station signing off is attached to an application program and SIGN OFF is a SMCS command, the signal character must prefix this command.

If the EXCEPT option is specified in the JOBS file, the application program is notified that the user has signed off.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

<signal character>
The <signal character> is the current signal character of the station.

CLEAR
If the CLEAR option is specified, the station is detached and cleared from the file information block of the remote file.

Examples:

I: *off
O: SIGNED OFF "CANDE"

I: *sign off
O: SIGNED OFF "CANDE"

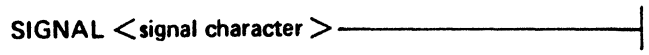
I: *sign off clear
O: SIGNED OFF "CANDE"

## SIGNAL

The SIGNAL command changes the signal character used by stations to communicate with the SMCS program. The default value for the signal character is the asterisk (*) character. The change takes place when no users are logged on to the SMCS program or signed on to other programs.

This command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

<signal character>
A single character except a blank or a period (.) character used to represent the new signal character. Refer to appendix B for a list of the allowable signal characters.

Example:

```
I: signal $
O: SIGNAL CHARACTER CHANGED TO '$'
```

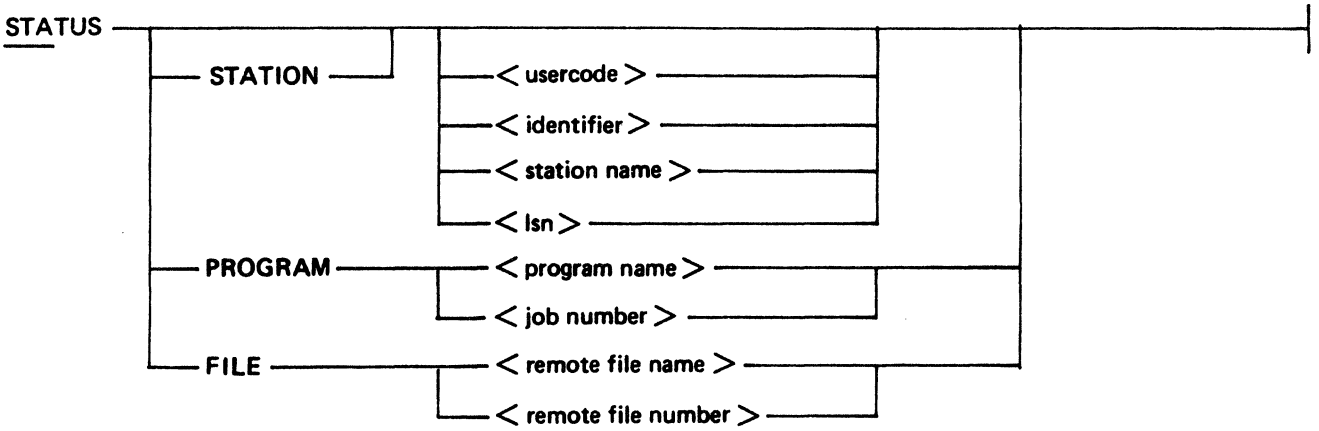
## STATUS

The STATUS command displays the current status of a program, file, or station.

When the key word is omitted from the command string, STATION is assumed. When the variable data is omitted from the command string, the originating station is assumed.

This command is valid from all stations.

Syntax:




Semantics:

<usercode>
The usercode of the station for which a status check is to be performed. The usercode is the usercode entered with the USER command.

<identifier>
The identifier of the station for which a status check is to be performed. The identifier is the identifier entered with the ID command.

<station name>
The symbolic station name of the station for which a status check is to be performed.

<lsn>
The logical station number (LSN) of the station for which a status check is to be performed.

<program name>
The name of the program for which a status check is to be performed.

<job number>
The job number assigned by the MCP of the program for which a status check is to be performed.

<remote file name>
The external file name of the remote file for which a status check is to be performed.

<remote file number>
The remote file number of the remote file for which a status check is to be performed.

Examples:

```
I: status program rd
I: status program 2832
O: PROGRAM "RD" (JOB NO = 2832):
        INPUT FILE = "RMT" (FILE NO 003) OPEN

I: Status file rmt
I: Status file 3
O: FILE "RMT" (REMOTE FILE NO = 003) (SIMPLE HEADERS):
        OPEN FOR 'RD' (INPUT,OUTPUT)
        002 OF 016 STATION(S) ASSIGNED
        PROTOCOL=02 NO.EOF=TRUE NO.SCROLL=FALSE
        LOG.ON=TRUE EXCEPTION.OK=TRUE RESTRICTIONS=TRUE
        PASS LSN'S: (3,4)
        GOODRESULTS OFF, CONTROLLED BY THIS MCS
I: status
I: status platypus
I: status station 4
O: STATUS FOR STATION "PLATYPUS" (LSN= 004):
        UNASSIGNED
        ATTACHED TO THIS MCS; SIGNAL CHARACTER = @
        READY, ENABLED, VALID
        TERMINAL TYPE = TD832
        BUFFERSIZE = 2250
        MYUSE = INPUT/OUTPUT
        ADDRESS (R/T) = PL/PL, TRAN = 0/0
        FREQUENCY (R/T) = 250/250
        RETRIES = 255, OUTPUT MESSAGES = 0
        LOGICALACK OFF, DIAGNOSTIC REQUEST OFF, GOODRESULTS OFF
        PRIMARY FILE = 002 (HEADERS)
        NO SECONDARY FILE ASSIGNED
        STATION PHONE = NULL
        CURRENT LINE NUMBER = 2
        NUMBER OF LINES = 1
        LINE INFORMATION (LINE #/DIALOUT STATUS) = 02/0
        TRANSLATE ON, AUDIT OFF
```

## STOP

The STOP command terminates the SMCS program in an orderly manner. It prevents any further SIGN ON, ATTACH, PASS, USER, or other active commands from being processed. A message is sent to all stations requesting that the users log off as soon as possible. After sending the message(s), the SMCS attempts to terminate every 30 seconds and automatically terminates after the last user logs off.

This command is valid only from the system ODT.

System ODT syntax:

Example:

```
I: stop
O: TERMINATION IN PROGRESS
```

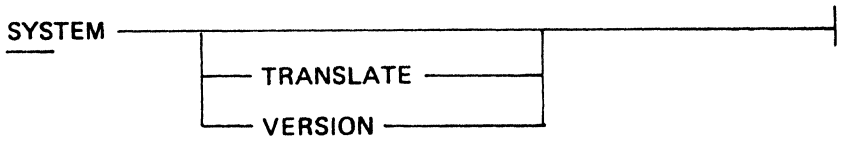
## SYSTEM

The SYSTEM command causes the current setting of the system translate option, the current version of the SMCS program, and the system Hostname to be displayed.

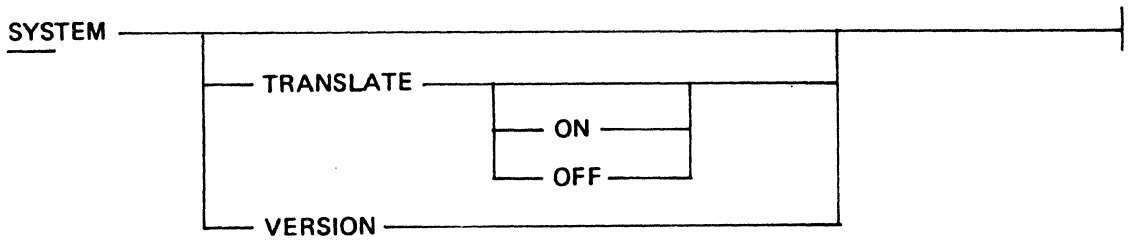
A remote ODT station or the system ODT can change the current setting of the system translate option.

If the TRANSLATE or VERSION keywords are omitted, the current value of the system Hostname, the current setting of the system translate option, and the SMCS version are displayed.

Workstation and controller syntax:




Remote ODT and system ODT syntax:




Semantics:

TRANSLATE
This option displays the current setting of the system TRANSLATE option.

ON
This option sets the system TRANSLATE option.

OFF
This option resets the system TRANSLATE option.

VERSION
This option displays the current version of the SMCS program.

Examples:

```
I: sys
I: system
O: HOST NAME: NIRVANA
     SYSTEM-TRANSLATE: ON
     SYSTEM-VERSION: B1000 SMCS MARK 10.0.007 (08/26/82 11:23)
I: system version
O: SYSTEM-VERSION: B1000 SMCS MARK 10.0.007 (08/26/82 11:23)
I: system translate on
O: SYSTEM-TRANSLATE: ON
```

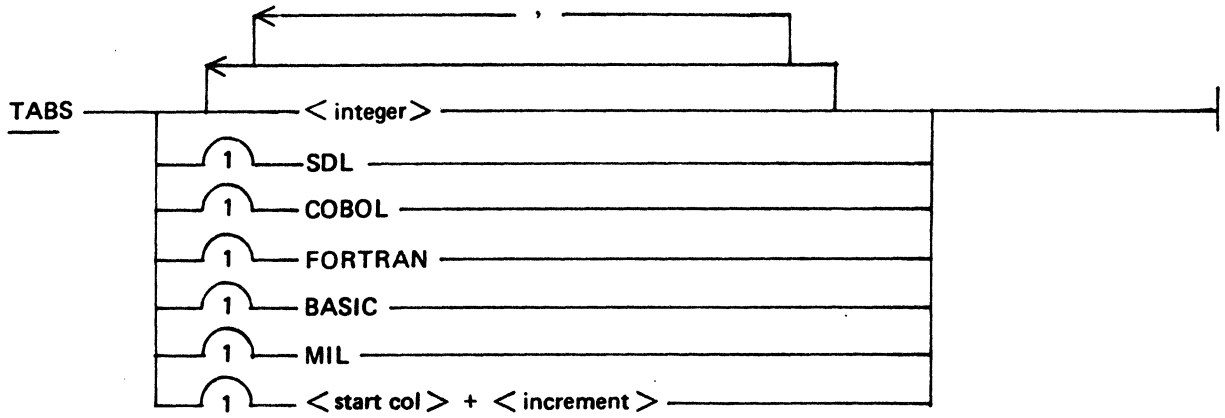
## TABS

The TABS command sets the tab stops on a TD 830, TD 820, MT 983 or MT 985 terminal.

NOTE
The variable tab option must be enabled in the terminal for this command to function. Refer to the TD 730/TD 830 System Reference Manual, and the MT 983/MT 985/MT 993 Display Terminals Reference Manual for instructions on enabling the variable tab option in the TD 830 and MT 983/985 terminals. For the TD 820 terminal, consult a Burroughs Field Engineer.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

integer
This field specifies the column number of the tab stop to be set on the terminal and can be repeated as needed. The <integer> token must be greater than zero and less than 81.

SDL
This keyword sets the tab stops at columns 9, 13, 18, 23, 28, 33, 38, 43, 48, 53, 58, 63, 68, 73, and 78 (column 9 and every five columns thereafter).

COBOL
This keyword sets the tab stops at columns 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, and 73 (column 8 and every four columns until column 48 and column 73).

FORTRAN
This keyword sets the tab stops at columns 9, 15, 19, 23, 27, 31, 35, 39, 43, 47, and 75 (columns 9 and 15 and every four columns until column 47 and column 75).

BASIC
This keyword sets the tab stops at columns 1, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, and 80 (columns 1 and 6 and every four columns until column 46 and column 80).

MIL
This keyword sets the tab stops at columns 9, 15, and 48.

<start col>
The <start col> token indicates the column on the terminal at which the tabs stops are to be set. The <start col> token must be greater than 0 and less than 80.

<increment>
The < increment > token specifies the increment by which the tabs stops are to be set on the terminal starting at < start col > . The < increment > must be greater than 0 and less than (80 - < start pos > ).

**Examples:**

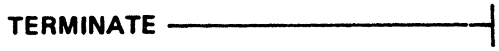
```
I: tabs cobol
I: tab fortran
I: tabs 9 13 20 30 40 50 60 70 80
I: tabs 10+10
O: #
```

## TERMINATE Command

The TERMINATE command terminates the step macro procedure that is currently running. Refer to section 6, Macro Procedures, for an explanation of the step macro procedure facility.

The TERMINATE command is valid from all stations except the system ODT.

Workstation, Remote ODT, and Controller syntax:




Example:

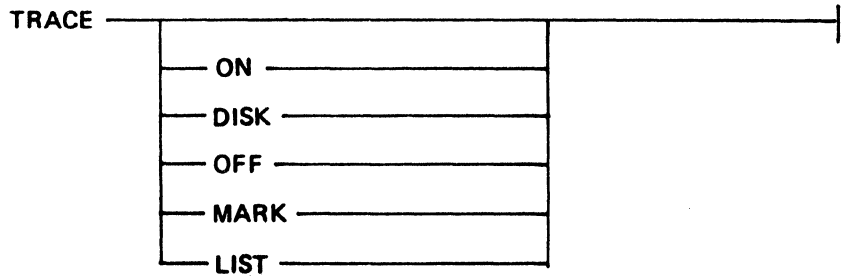
```
I: terminate
O: STEP MACRO TERMINATED
```

## TRACE

The TRACE command invokes the tracing facilities for SMCS input/output messages. If no option is specified, the current status of the TRACE option is displayed.

This command is only valid from a controller station or the system ODT.

Controller and system ODT syntax:




Semantics:

ON
Initiates printer tracing.

DISK
Initiates disk tracing.

OFF
Stops tracing.

Mark
Writes a marker message into the TRACE file that enables the user to locate significant events in the TRACE file.

LIST
Initiates a DMPALL listing of the disk TRACE file.

Examples:

I: trace
O: CURRENT TRACE STATE = ON/DISK (215)

I: trace on
O: PRINTER TRACE STARTED

I: trace off
O: TRACING STOPPED

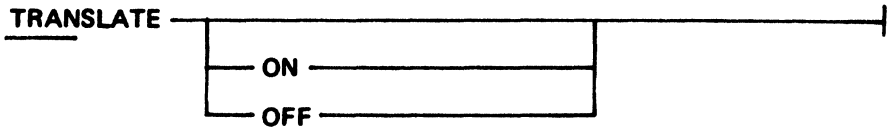
I: trace mark
O: TRACE FILE MARKED (12)

## TRANSLATE

The TRANSLATE command causes lowercase characters received by the SMCS program to be translated to uppercase characters.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

OFF
Disables the translation of messages to uppercase letters.

ON
Enables the translation of messages to uppercase letters.

Examples:

I: translate
O: CURRENT TRANSLATE STATE = SET

I: trans on
O: TRANSLATE SET

I: tran off
O: TRANSLATE RESET

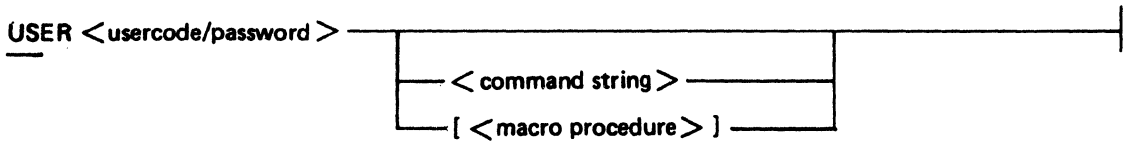
## USER

The USER command logs a user onto the SMCS program. Once the command statement is entered, the usercode and password are validated and stored. This information is then used for all MCP command strings sent directly to the MCP by the SMCS program as well as for station identification. The command string is optional; however, if one is entered, it is taken as a normal input message to the SMCS after the log-on sequence is completed. If usercode and password is already logged on at another station, the current log-on is allowed and the user is advised that other stations are already logged on under the same usercode.

The USER command is not scanned for macro procedures delimited by the bracket characters ([ ]). This means that the macro procedure invocation is not expanded before the USER command is processed and allows a user to log on with a macro procedure following the USER command and have it processed after the log-on is complete.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

<usercode>
A valid usercode.

<password>
A valid password.

<command string>
Any valid SMCS command.

<macro procedure>
This field can be any valid macro procedure for the usercode of the station. Refer to section 6, Macro Procedures, for more information on macro procedures.

Examples:

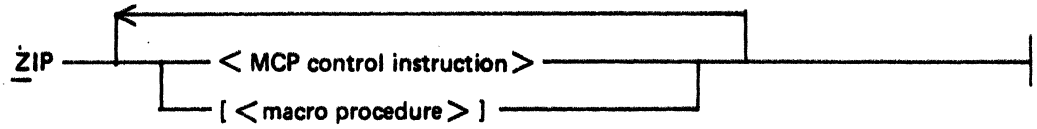
```
I: user apteryx/kiwi
I: user apteryx/kiwi on cande
I: user apteryx/kiwi [on]
O: B1000 SMCS MARK 10.0.007 (08/26/82 11:23) SYSTEM-ID: NIRVAN
   (APTERYX) LOGGED ON -SMCS-AT 09:14 01/06/83
   YOU ARE AT STATION 54 (TD832)

    <macro command response>

    <program log-on message>
```

## ZIP

The ZIP command sends the MCP a control instruction. The string is not made secure, meaning that the MCP control is not prefixed with the usercode/password pair of the station and that the RR attribute is set to 0 when it is spawned by the SMCS program. The logical station number (LSN) of the station is used as the session number.

This command is valid only from a remote ODT station.

Remote ODT syntax:




Semantics:

<MCP control instruction>
An MCP control instruction.

<macro procedure>
This field can be any valid macro procedure for the usercode of the station. Refer to section 6, Macro Procedures, for more information on macro procedures.

Examples:

```
I: zip us my/user ex dmpall
I: z ex dmpall
I: z [cobol p test]
I: zip us apteryx/kiwi [cobol p test]
O: <MCP response>
```

When zipping a secured string from a remote ODT station using the ZIP command, any responses to the job must be prefixed with the usercode with which the job was zipped. The following is an example of the correct and incorrect methods to respond to a job that has been spawned using the ZIP command and a usercode.

Correct

```
ZIP US A/B EXECUTE DMPALL;
ZIP US A/B <job #> AX <command>
```

Incorrect

```
ZIP US A/B EXECUTE DMPALL;
ZIP <job #> AX <command>
```

In the "incorrect" example, the MCP responds with an INVALID JOB NUMBER message.

## ? <MCP control instruction>

The MCP control instruction is spawned through a ZIP command to the MCP.

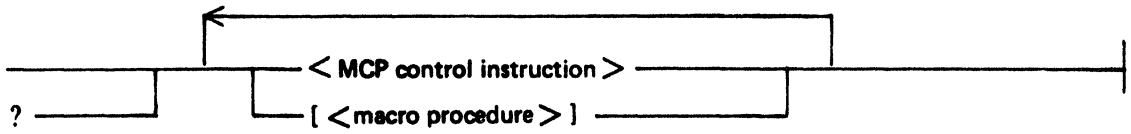
If the user is logged on to the SMCS program, the MCP control instruction is spawned using the usercode and password of the station, with the RR attribute set to 1, and the logical station number (LSN) of the station as the session number. The string is secured, meaning that ? MX returns jobs spawned with the usercode of the station only (if the RD program is not currently running in the mix). If a ?EXECUTE <program name>) control string is entered, <program name> is spawned through a ZIP command prefixed with the usercode and password of the station, the RR attribute set to 1, and the LSN as the session number. No reference is made to the JOBS file. Refer to the B 1000 Systems System Software Operation Guide, Volume 1, for more information on the RR attribute, session numbers, and job spawning. Refer to appendix C for a list of the valid MCP control instructions permitted from a workstation.

If a remote ODT station is not logged on to the SMCS program with a usercode, the MCP control instruction is not made secure, meaning it is spawned through a ZIP command without a usercode/password and with the RR attribute set to 0. The LSN of the station is used as the session number.

When a MX or WY MCP command is entered, the SMCS program first determines if the RD program is executing and, if so, if it can attach another station. If these conditions are true, the request is forwarded to the RD program for a response; otherwise, the SMCS program requests the function of the MCP.

This command is valid from all stations except the system ODT.

Workstation, remote ODT, and controller syntax:




Semantics:

<MCP control instruction>
Any MCP control instruction allowed from a workstation. Refer to appendix C for a complete list of restricted MCP control instructions.

<macro procedure>
This field can be any valid macro procedure for that usercode of the station. Refer to section 6, Macro Procedures, for more information on macro procedures.

Examples:

```
I: ? PD =/=
I: ex dmpall
I: [COBOL P PROGRAM]
O: <MCP response>
```

# SECTION 6
# MACRO PROCEDURES

The invocation of macro procedures enables the user to invoke lengthy MCP control strings or enter SMCS commands without having to enter the entire string or command. This is especially useful with control strings that require numerous file equation statements, or with a series of control strings that are to be executed sequentially. A macro invocation consists of a macro name followed by optional parameters and their replacement strings enclosed in the bracket characters ([ ]). After the macro procedure has been expanded by the SMCS program, the control string is treated as though the user had entered it directly. A macro invocation can appear in a control string zipped from a workstation, controller station, or a remote ODT station. Macro invocations can also appear in unsecured control strings invoked from a remote ODT station.

There are two types of macro procedures: (1) regular macro procedures and (2) step macro procedures. For discussion purposes, regular macro procedures are referred to simply as macro procedures.

The commands in a regular macro procedure are handled asynchronously; that is, they are spawned in one zip-string.

The commands in a step macro procedure are handled synchronously; that is, they are processed one at a time. Action taken is dependent on the outcome of the previous step in the step macro procedure. Refer to the discussion in this section on step macro procedures.

## MACRO PROCEDURE CREATION

Macro procedures are created using the MACRO command or can reside in a file under usercode of the station. Refer to SMCS Input Commands in section 5 for information on the MACRO command.

## MACRO PROCEDURE INVOCATION

A macro procedure is invoked by enclosing the name of the macro procedure in square bracket characters ([ ]) along with any parameters and their replacement strings. The first token after the left bracket character ([) is the macro name.

When the left bracket character ([) is encountered, the SMCS program performs the following steps:

1. The SMCS program first searches for the macro procedure in the DIRECTORY file under usercode or identifier of the station. If the macro procedure is found, the SMCS program procedes to step 2; otherwise, the following is done:

   1) If the macro procedure is a step macro procedure, an error message is generated.
   2) The SMCS program attempts to open a disk file using the macro name and the usercode of the station. If the file is not present, the bracket ([ ]) characters are ignored and the string is spawned to the MCP.

2. If the macro procedure is present, the SMCS program scans the macro procedure for the parameters. If a parameter is found, it is replaced by the replacement string. The SMCS program then concatentates the text portion of the records in the macro procedure and removes extra blank characters to produce a text string. If a percent sign (%) character is found in a record, the remaining text in the record is not scanned. The resulting text string is then spawned to the MCP or the controlling remote file depending on the nature of the text string.

The following are examples of macro procedures and the resultant control strings, with and without parameters.

Example 1 (without parameters):

Macro procedure MIX contains:

```
ML;
MX;
WY;
WS =/=;
```

Entering *Z [MIX] causes the following control string to be sent to the MCP:

```
ZIP QU SMCS/MCPQ LS SZ 21 RR 0 ML; MX; WY; WS =/=;
```

The result is returned to the terminal from which the command originated.

Example 1 shows how a user can invoke often-used combinations of ODT commands with one command. However, compile and execute strings usually use a standard format with a file name substituted throughout the file equations. This type of operation can be accomplished by using macro parameters.

Example 2 (with parameters):

The macro procedure COBOL contains:

```
COMPILE P COBOL LIBRARY;
        FILE CARDS DSK NAME P.S;
        FILE LINE U.N NO HAR NO ATP NAME #P;
```

Entering * [COBOL P TEST] causes the following control string to be zipped to the MCP:

```
ZIP QU SMCS/MCPQ LS SZ 21 RR 1 US WOMBAT/ COMPILE TEST COBOL
LIBRARY; FILE CARDS DSK NAME TEST.S; FILE LINE U.N NO HAR
NO ATP NAME #TEST;
```

In example 2, the parameter is the token P and the replacement string is TEST. The period (.) character in the token, P.S, is used as a delimiter for a token search. The valid delimiters are:

```
<space> = "" ( ) # / ; % __ . * , ! -[ ]
```

When these delimiters are encountered by the SMCS program, they are left in the control string. A special delimiter exists that is not put in the control string. This is the !! delimiter. In the second example, if the token P.S was P!!S, the replacement string is TESTS instead of TEST.S.

NOTE

The vertical bar character (|) is created by entering an exclamation point (uppercase 1) on a MT 985/MT 983, TD 830, or TD 820 terminal. It is represented by the EBCDIC code @4F@.

Combinations of multiple macro invocations and text can be zipped in one control string. For example, using the macro procedure in example 2, the following can be invoked:

```
* WT; [COBOL P TEST1]; THEN [COBOL P TEST2];
```

This produces the following control string:

```
ZIP QU SMCS/MCPQ LS SZ 21 RR 1 US WOMBAT/ WT; COMPILE TEST1
COBOL LIBRARY; FILE CARDS DSK NAME TEST1.S; FILE LINE
U.N NO HAR NO ATP NAME #TEST1; THEN COMPILE TEST2 COBOL
LIBRARY; FILE CARDS DSK NAME TEST2.S; FILE LINE U.N NO
HAR NO ATP NAME #TEST2;
```

A macro invocation can invoke another macro procedure. For example, the macro procedure C, contains the following:

```
COMPILE X Y LIBRARY;
FILE CARDS NAME X!!SRC;
```

The macro procedure CMPPAY contains the following invocation of the macro procedure C:

```
[C X PAYROLL Y COBOL]
```

The invocation of the macro procedure CMPPAY, [CMPPAY], produces the following MCP control string:

```
ZIP QU SMCS/MCPQ LS SZ 21 RR 1 US WOMBAT/ COMPILE PAYROLL
LIBRARY; FILE CARDS DSK NAME PAYROLLSRC;
```

Macro procedures can also invoke SMCS commands. For example, if the macro procedure SETTABS contains the following:

```
TABS 9 20 30 40 50 60 70 80
```

Then the invocation of the macro procedure SETTABS, [SETTABS], causes the tab stops on a terminal to be set. Using the TABS command in a macro procedure can be especially useful to the RPG programmer.

# STEP MACRO PROCEDURES

Step macro procedures proceed programmatically through a series of steps and take action depending on the outcome of each step.

Step macro procedures are added, changed, or deleted using the MACRO command. When entering step macro procedures using the MACRO command, column 80 must be non-blank to signify that it is part of a step macro procedure. Double pound sign (##) characters indicate a step macro command.

A step in a step macro procedure can be a record destined for the MCP, remote file, or the SMCS program itself. After the SMCS program recognizes a step of a step macro procedure, the following is checked:

1. If the first character is not a signal character and this station is attached, or in the process of being attached, to a remote file, then the text is given to the remote file as normal input.
2. If the first character is the signal character or if no remote file is attached to this station, then the SMCS program acts on this step. It checks columns one and two (two and three if a signal character was present) for the double pound sign (##) characters. If no match is found, the SMCS program treats the record as any other input from a terminal. That is, the SMCS program checks to see if the record is an SMCS command and acts upon it if it is; otherwise, the SMCS program spawns the record to the MCP. If the double pound sign (##) characters were found, the SMCS program expects one of the step commands to follow on the same record.

# STEP MACRO COMMANDS

The step macro commands control programmatically the flow of the step macro procedure.

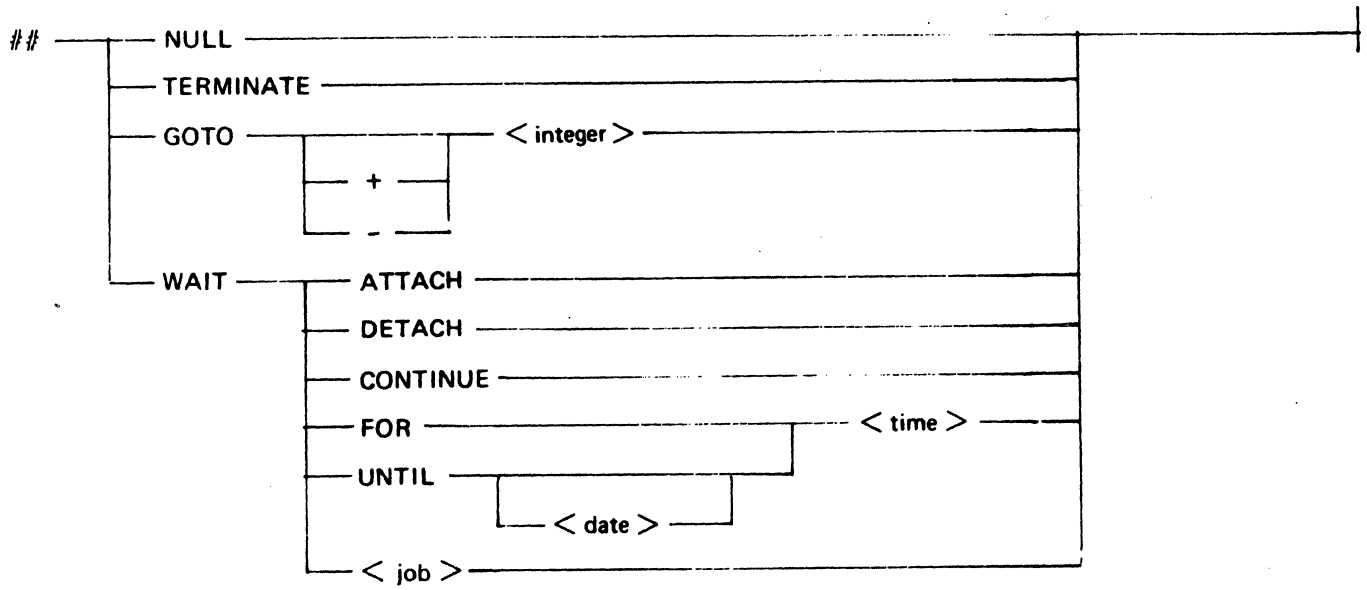
If a step of a step macro procedure is not a WAIT statement, the step macro procedure is not suspended and the next step is read. The step macro procedure is terminated after the last record is processed if that record is not a GOTO.

A signal character is used, as a step in the step macro procedure, when the user wants the step to be processed by the SMCS program, and there is a chance that the station has become attached to a remote file and the step is being sent to that remote file.

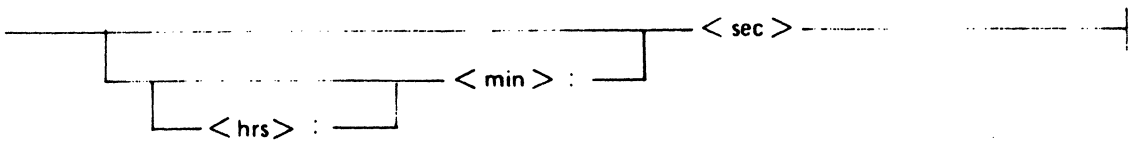
A step macro procedure cannot be contained in a disk file whereas a macro procedure can.

The WAIT FOR and WAIT UNTIL commands are stored FOR the time specified or UNTIL the date and/or time specified arrives, even if the SMCS program has been restarted.

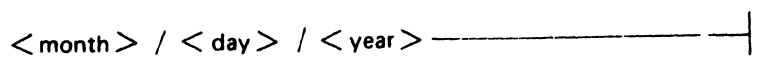
Syntax:




The format for the <time> is:




The format for the <date> is:



Semantics

NULL
The NULL command causes the next step in the step macro procedure to be read.

TERMINATE
The TERMINATE command causes the step macro procedure to be terminated.

GOTO
The GOTO command causes control to go to the indicated record (relative to 1) as the next step in the step macro procedure. Constraints are placed on the use of the GOTO step macro command so that the step macro procedure does not enter a loop. The GOTO command works effectively when used in conjunction with the WAIT job option.

WAIT
If a WAIT command is performed, processing of the step macro procedure is suspended. When the WAIT event comes true, the SMCS program starts at the next step of the step macro (except for WAIT job).

WAIT ATTACH
The WAIT ATTACH command causes the step macro procedure to be suspended until the SMCS program becomes the secondary remote file (for example, until the station becomes attached to another program). If the SMCS program is currently the secondary remote file, this command is treated as a NULL statement.

WAIT DETACH
The WAIT DETACH command causes the step macro procedure to be suspended until the SMCS program becomes the primary remote file (for example, until the station is detached from any program currently running). If the SMCS program is currently the primary remote file, this command is treated as a NULL statement.

WAIT CONTINUE
The WAIT CONTINUE command causes the step macro procedure to be suspended until the SMCS CONTINUE program command is entered at the station.

WAIT FOR
The WAIT FOR command causes the step macro procedure to be suspended for a <time> period of time.

WAIT UNTIL
The WAIT UNTIL command causes the step macro procedure to be suspended until the date specified by < date > and/or the time specified by < time > . For example, the statement

```
*## WAIT  UNTIL  14:30   06/01/83
```

causes the step macro procedure to be suspended until 2:30 P.M. on June 1, 1983.

WAIT job

If the token following the WAIT keyword is not the ATTACH, DETACH, CONTINUE, FOR, or UNTIL keywords, the token is taken as a control string to be spawned to the MCP. The control string can be either a MCP control instruction string or a macro procedure. If the control string is a macro procedure, it is expanded before the string is spawned. If the control string causes at least one job to be executed, the job number of the first job to be executed is stored; any others are ignored. The step macro procedure is suspended until the job goes to end of job. When the job goes to end of job, the step macro procedure is restarted at:

1. The next record after the WAIT job, if the end of job was normal with no syntax errors.
2. The next record + 1 if it was an abnormal end of job (for example, the program aborted).
3. The next record + 2 if it was a normal end of job but had syntax errors.

<hrs>

The <hours> token is the number of hours that the step macro procedure is to wait before restarting. The <hrs> token must be an integer value.

<min>

The <min> token is the number of minutes that the step macro procedure is to wait before restarting. The <min> token must be an integer value between 0 and 59.

<sec>

The <sec> token is the number of seconds that the step macro procedure is to wait before restarting. The <sec> token must be an integer value between 0 and 59 if the <min> and/or <hours> token are specified. If the <min> and/or <hours> token are not specified, the <sec> token must be an integer value.

<month>

The <month> token is the month portion of the date in the WAIT UNTIL keyword. The <month> token must be an integer value between 1 and 12.

<day>

The <day> token is the day portion of the date in the WAIT UNTIL keyword. The <day> token must be an integer value between 0 and 31.

<year>

The <year> token is the last two digits of the year portion of the date in the WAIT UNTIL keyword.

Examples:

```
MACRO ADD CYCLE
*OFF                %   1. Log off any program you might be on.
*##WAIT DETACH      %   2. Wait for the detach to occur.
ON CANDE G X        %   3. Log on to CANDE and get the source.
P 0                 %   4. Get a page of source.
*##WAIT CONTINUE    %   5. Wait till I stop making changes.
SAVE                %   6. Issue a SAVE.
BYE                 %   7. Issue a BYE.
*##WAIT DETACH      %   8. Wait for the detach to occur.
*[Y X]              %   9. Issue macro Y with source X.
*[RD]               %  10. Return to RD.
```

The invocation of the step macro procedure, CYCLE, is

```
[CYCLE X SOURCE Y <MACRO>].
```

If the above step macro procedure is repetitively executed, another macro procedure can be added as follows:

```
MACRO ADD CC
[CYCLE X SOURCE-NAME Y [DCTR] 0 1700]
```

All that is necessary to invoke the step macro procedure is:

```
*[CC]
```

Another example of a step macro procedure involving a compile follows:

```
MACRO ADD BC
*OFF                    %   1 Log off whatever you might be on.
*##WAIT DETACH          %   2 Wait for detach or a BYE from RL.
ON CANDE G X!!SRC       %   3 On to CANDE and get source file.
P C                     %   4 Get a page.
*##WAIT CONTINUE        %   5 Wait for CONTINUE command.
SAVE                    %   6 Issue SAVE.
BYE                     %   7 Issue BYE.
*##WAIT DETACH          %   8 Wait for detach.
[RD]                    %   9 Back to RD. I leave my terminal on RD.
*##WAIT [C]             %  10 Suspend macro until compile is done.
*##TERMINATE            %  11 NORMAL EOJ
*##NULL                 %  12 ABNORMAL EOJ DS DP. Fall thru to next step.
*OFF                    %  13 SYNTAX ERRORS. Log off RD.
*##WAIT DETACH          %  14 Wait for detach.
EX RL                   %  15 Execute RL.
*##WAIT ATTACH          %  16 Wait for RL to become attached.
G X!!BKP                %  17 Get the backup file.
ERR                     %  18 Look for errors.
*##GOTO 2               %  20 Go back and wait for the BYE.
```

Records 10 through 13 in the above example are an example of the WAIT job option. The invocation of this step macro procedure would be:

```
[BC X TST].
```

All occurrences of X in the step macro procedure are replaced with TST, the source file name. It should also be noted that the replace operation takes place on any internal macro invocations.

## LIMITS

The following limits exist in creating and invoking macro procedures:

1. The length of the macro procedure entered by the MACRO ADD or the MACRO CHANGE command can be up to 23 lines.
2. The length of the control string spawned as a result of a macro invocation can be up to 1000 characters.
3. There can be up to 10 parameters in each macro procedure.
4. The SMCS program does not scan for macro procedures invocations in the ID, LOGON, MACRO, PASS, or USER commands.

   This means that the macro invocation is not expanded before the USER or ID command is processed. This allows a user to log on with a macro procedure following the USER or ID command and have it processed after the log on is complete.

   Example:

   USER A/B [ON] % "ON" is a macro procedure

5. A parameter may be up to 5 characters in length.
6. The replacement field of a parameter may be up to 15 characters in length.

# SECTION 7
# SYSTEM OUTPUT MESSAGES

The three types of output messages are listed below in the order in which they are presented:

Master Control Program messages
Network controller messages
SMCS Program messages

## MASTER CONTROL PROGRAM MESSAGES

The Master Control Program (MCP) displays messages in response to a control instruction or to inform the system operator of events that have taken or are to take place. All MCP output messages are displayed on the system ODT and start in column two. Refer to the B 1000 Systems System Software Operation Guide, Volume 1 for further information regarding the MCP output messages.

## NETWORK CONTROLLER MESSAGES

The messages displayed by the network controller on the system ODT start in column two and have the following format:

% <network controller name> = <job number> <message>

## SMCS OUTPUT MESSAGES

The SMCS program output messages, whether output to a remote terminal or to the system ODT, are either informational, warning, error, or fatal error messages.

### Informational Messages

Informational messages result from the processing of commands and system message traffic and in most cases do not require an operator response.

### Warning Messages

A warning message is an indication of a temporary SMCS condition that has taken place. The problem has been resolved by the SMCS and the warning message is displayed to indicate the condition.

### Error Messages

Error messages are displayed in response to an improper use of an SMCS command. These commands are rejected by the SMCS program and an error message reply is displayed.

## Fatal Error Message

A fatal error message indicates that the SMCS program has encountered a problem that it cannot resolve. The fatal error message is displayed on the system ODT. When this error condition occurs, it is necessary to shut down the SMCS program in an orderly manner; however, the system operator has the option to record any information that can be useful before that action is taken. After a fatal error, the SMCS program displays an ACCEPT message. To terminate the SMCS program in an orderly manner, the system operator enters a blank response to the SMCS ACCEPT message in the following manner:

```
<SMCS job number>AX
```

A SMCS program memory dump is automatically produced by the SMCS program prior to termination.

# OUTPUT MESSAGES

The output messages that can occur while executing the SMCS program are shown and described on the following pages.

'<program id>' NOT STARTED – COPY LIMIT EXCEEDED
The COPIES option is specified in the JOBS file but the maximum number of copies permitted is currently running.

'<program id>' NOT STARTED – STATION USERCODE NOT IN JOBS FILE
The USERCODE option is specified in the JOBS file and the usercode of the station does not match the one specified.

'<program id>' NOT STARTED – STATION USERCODE NOT IN USERFILE
The USERFILE option is specified in the JOBS file and the usercode of the station was not specified in the userfile.

'<program id>' NOT STARTED – USERCODE EXPECTED
FOLLOWING USERCODE OPTION IN JOBS FILE
A usercode was not specified in the JOBS file following the USERCODE option.

'<program id>' NOT STARTED – USERCODE CANNOT BE
SPECIFIED IN JOBS FILE WHEN USERFILE OPTION IS
Both the USERCODE and USERFILE options were specified in the JOBS file for the same job.

'<program id>' NOT STARTED – USERFILE CANNOT BE
SPECIFIED IN JOBS FILE WHEN USERCODE OPTION IS
Both the USERCODE and USERFILE options were specified in the JOBS file for the same job.

'<program id>' NOT STARTED – VALID FILE NAME
EXPECTED FOLLOWING USERFILE OPTION IN JOBS FILE
An invalid file name was specified in the USERFILE option in the JOBS file for the job.

'<program name>' TERMINATED (NORMAL), STATION SIGNED OFF
The program to which the station was signed on has terminated. The station has been detached and is currently attached to the SMCS.

CANNOT PASS TO PROGRAM <program id> – STATION USERCODE NOT IN JOBS FILE
The USERCODE option was specified in the JOBS file for this job and the usercode of the station does not match the usercode specified.

CANNOT PASS TO PROGRAM <program id> – STATION USERCODE NOT IN USERFILE
The USERFILE option was specified in the JOBS file for this job and the usercode of the station was not present in the userfile.

CANNOT SIGN ON PROGRAM <program id> – STATION USERCODE NOT IN JOBS FILE
The USERCODE option was specified in the JOBS file for this job and the usercode of the station does not match the usercode specified.

CANNOT SIGN ON PROGRAM <program id> – STATION USERCODE NOT IN USERFILE
The USERFILE option was specified in the JOBS file for this job and the usercode of the station was not present in the userfile.

CONTINUE COMMAND NOT VALID
The CONTINUE command was entered at the station and the station did not have a step macro procedure in process that was waiting on a CONTINUE command.

ENDING QUOTE MISSING IN JOBS FILE
A quoted item in the JOBS file was missing the ending quotation mark (").

INVALID STEP MACRO COMMAND
An invalid step macro command was entered following the ## characters in the step macro procedure.

INVALID WAIT MACRO COMMAND
A wait on an invalid event was specified in the ## WAIT step macro command.

JOB "<program id>" NOT AUTO-STARTED – USERCODE AND USERFILE OPTIONS CANNOT BOTH BE SPECIFIED IN JOBS FILE
The specified job is not auto-started because both the USERCODE and USERFILE options were specified for the job in the JOBS file.

LOG-OFF IS REQUIRED – ENTER BYE
The ID command was entered while a user at a station which was not declared as a remote ODT station was logged on to the SMCS program with the USER command.

LOG-ON WITH A VALID USERCODE/PASSWORD IS REQUIRED
A user at a station which is not declared as a remote ODT station must log on before entering any command other than the HELP, ID, REPORT, SEND, or TRANSLATE command.

MACRO ADDED
The macro procedure specified by the MACRO ADD command has been added to the DIRECTORY file.

MACRO CHANGED
The macro procedure specified by the MACRO CHANGE command has been changed in the DIRECTORY file.

MACRO DELETED
The macro procedure specified by the MACRO DELETE command has been deleted from the DIRECTORY file.

MACRO STEP COMMAND EXPECTED
No step macro command was found following the ## characters in the step macro procedure.

MESSAGE QUEUE FULL – MACRO TERMINATED
The SMCS program attempted to deliver a message from a step macro procedure to a program whose input queue was full. The step macro procedure is terminated.

MISSING ";" IN JOBS FILE
The required semicolon was not found while scanning the JOBS file for a program-id.

MISSING CLOSE PAREN FOR TIMER OPTION IN JOBS FILE
The NO-EOF option in the JOBS file was missing a close parenthesis character " ) ".

MISSING CLOSE IN JOBS FILE
A close parenthesis character " ) " was missing in the JOBS file.

MUST CREATE (MAIL/<day name>) FILE, PLEASE SOLICIT ITS RELEASE
The MAIL file for today is currently in use by a program. It must be released before the SMCS program can use it. This message is displayed on the system ODT.

NON-NUMERIC VALUE ENTERED FOR NO-EOF OPTION IN JOBS FILE
A non-numeric value was specified for the timer value in the NO-EOF option in the JOBS file.

PROGRAM QUEUE FULL – PLEASE SEND MESSAGE LATER
The SMCS program attempted to deliver a message from a station to a program whose input queue was full.

QUEUE FULL, MESSAGE(S) LOST
This message is to inform the user at the terminal that an undetermined number of output messages were not processed by the MCP.

REMOTE FILE CLOSED BY <program name>, STATION SIGNED OFF
This message indicates that the program to which the user was signed on terminated. The SMCS notifies all stations attached to that program of this termination.

SMCS TERMINATING: PLEASE RELEASE THIS STATION ASAP
This message is broadcast to all active stations when a STOP command is entered at the system ODT. The SMCS program is terminating.

STATION <station name> OFFLINE
This message indicates that the station identified is considered off-line by the network controller.

STEP MACRO TERMINATED
The currently invoked step macro procedure has been terminated.

TERMINATE COMMAND NOT VALID
The TERMINATE command was entered and the station did not have a step macro procedure currently running.

UNKNOWN OPTION "<identifier>" SPECIFIED IN JOBS FILE
An identifier which was not a valid option was scanned in the JOBS file.

USERCODE EXPECTED FOLLOWING USERCODE OPTION IN
JOBS FILE FOR JOB <program id>
An identifier which was not a valid option was scanned in the JOBS file.

VALID FILE NAME EXPECTED FOLLOWING USERFILE OPTION
IN JOBS FILE FOR JOB <program id>
The USERFILE option was specified in the JOBS file for this job with an invalid file name. The job is not started.

VALID US/PW REQUIRED FOR USERFILE OPTION IN JOBS FILE
An invalid usercode and password was specified to access the userfile for the USERFILE option in the JOBS file.

**** END MACRO LIST ****
Signifies the end of the macro procedure listed by the MACRO LIST <macro name> command.

> WARNING: MESSAGE QUEUE FULL, SEND MESSAGE LATER
When the user signed on to a specified program, a message string followed the program name. This message could not be passed along to the program nor could it be saved in the program message queue; therefore, it was dropped. When the sign-on is actually completed, the message string must be entered directly to the program.

> WARNING: MESSAGE ERROR ENCOUNTERED; NO MESSAGES REMOVED
The SMCS encountered an error when attempting to access the messages to be removed. No further action has been taken.

>> ERROR: <STRING> REQUIRED
This message occurs when a character grouping representing a user identity was omitted from the command string.

>> ERROR: (SYSTEM)/USERCODE FILE IS NOT PRESENT.
The usercode/password file is not available to the SMCS program; the usercode/password file is not in the system directory.

>> ERROR: '<file name>' IS NOT ON DISK
The file name entered with the command string is not in the disk directory.

>> ERROR: "<program id>" NOT IN JOBS FILE
The <program id> specified in the SIGN ON or PASS command is not in the JOBS file.

>> ERROR: <text> LARGER THAN <number> CHARACTERS
The text portion of the macro procedure is larger than <number> characters.

>> ERROR: '<usercode>' IS NOT A VALID USER
This message is displayed when the slash character (/) and/or password have been omitted from the command string.

>> ERROR: 'CHANGE' DIAGNOSTIC REJECTED, INVALID.
This message means that the network controller is not prepared to handle this optional facility.

>> ERROR: 'CHANGE' LSN '<lsn>' NOT DONE: STATION BELONGS TO ANOTHER MCS
This means that a CHANGE command string was entered to the SMCS while the station was signed on to another program which is an MCS.

>> ERROR: 'PASS' REQUIRES '<PROGRAM NAME> <STRING>'.
The PASS command was entered without the appropriate information in the command string.

>> ERROR: 'REVERSE' IS THE ONLY VALID OPTION ON 'SCROLL'.
This message informs the user that the exact keyword option REVERSE was not entered properly.

>> ERROR: 'US' REQUIRES VALID <USERCODE>/<PASSWORD>.
This message indicates that a usercode did not prefix the slash character (/) in the command string.

>> ERROR: "ALL" OR "ONE" OR <NUMBER> EXPECTED
A keyword of ALL or ONE or an acceptable integer was not found following the MOVE command in the command string.

>> ERROR: "CHANGE" IS A CONTROLLER STATION COMMAND
The CHANGE command is not valid from a workstation or remote ODT station.

>> ERROR: "DUMP" IS A CONTROLLER STATION COMMAND
The DUMP command is only valid from a controller station.

>> ERROR: "DYNAMIC" IS A HOST ODT COMMAND
The DYNAMIC command is acceptable only from the system ODT.

>> ERROR: "FILE" MISSING
This message indicates that neither the keyword PROGRAM nor FILE was included in the command string; one or the other keyword is required.

>> ERROR: "JOBS" NOT ALLOWED FROM HOST ODT
The JOBS command is not allowed from the system ODT.

>> ERROR: "NEWS" IS A ODT COMMAND
The system NEWS message can only be entered from a remote ODT or system ODT.

>> ERROR: "PASS" NOT ALLOWED FROM HOST ODT
The system ODT operator is not allowed to PASS information to any user program.

>> ERROR: "PASS" TO (<program id>) REQUIRES USER LOGGED-ON
The PASS command is rejected because the program-id specified in the PASS string uses protocol type 1 which requires a usercode, and the user is not logged on.

>> ERROR: "STOP" IS A HOST ODT COMMAND
Entry of the STOP command is only allowed from the system ODT.

>> ERROR: "TRACE" IS A CONTROLLER STATION COMMAND
Tracing can only be initiated from a controller station or the system ODT.

>> ERROR: CAN ONLY CLEAR FILES. A STATION MAY BE 'DETACH'ED
The keyword FILE was omitted from the command string.

>> ERROR: CANNOT 'BYE' - CURRENTLY ATTACHED.
This message indicates that a command was entered to log off the SMCS while the station was attached to another program.

>> ERROR: CANNOT 'SCROLL REVERSE' - FORWARD SCROLL ENABLED
The station must not be in forward scroll mode to change terminal scrolling.

>> ERROR: CANNOT "LOG ON": SMCS TERMINATING
The SMCS is terminating; no more stations can log on.

>> ERROR: CANNOT "PASS": FILE TABLE FULL
The SMCS program does not have space available in the remote file table for a new program entry for the PASS command.

>> ERROR: CANNOT "PASS": PROGRAM TABLE FULL
The SMCS program table is full. The command string has been rejected and must be reentered.

>> ERROR: CANNOT "PASS": SMCS TERMINATING
The SMCS program is terminating.

>> ERROR: CANNOT "SIGN ON": FILE TABLE FULL
The SMCS does not have an available entry remaining in the remote file table to grant the sign-on.

>> ERROR: CANNOT "SIGN ON": PROGRAM TABLE FULL
The SMCS does not have an available entry in the program table for the program sign-on.

>> ERROR: CANNOT "SIGN ON": SMCS TERMINATING
The SMCS program is terminating.

>> ERROR: CANNOT "ZIP": STATION NOT A REMOTE ODT
The ZIP command is only permitted from a remote ODT.

>> ERROR: CANNOT (ZIP): STATION NOT CONTROLLED BY THIS <SMCS program id>
The station from which the command was entered is currently under the control of another MCS.

>> ERROR: CANNOT ACCEPT DOT AS SIGNAL CHARACTER
A period character (.) is disallowed as a signal character.

>> ERROR: CANNOT CHANGE "SIGNAL" FOR OTHER STATIONS
This error message occurs when an attempt is made to change the signal character of a terminal other than itself.

>> ERROR: CANNOT CHANGE "SIGNAL" UNLESS SIGNED ON
Terminal signal character changes are valid only when signed on to a program.

>> ERROR: CANNOT CHANGE SCROLL MODE FOR OTHER STATIONS
Only the scroll option on the station initiating the scroll change command can be changed.

>> ERROR: CANNOT INITIATE PROGRAM: <SMCS program id> TERMINATING
No commands are accepted or processed once the SMCS is terminating.

>> ERROR: COMMAND NOT PERMITTED FROM HOST ODT
The command is not valid at the system ODT.

>> ERROR: COMMAND OPTION RESTRICTED TO AN ODT
An attempt was made to change one of the system parameters from a station not declared as a remote ODT station.

>> ERROR: COMMAND RESTRICTED TO HOST ODT
The command is permitted only at the system ODT.

>> ERROR: DESTINATION NOT SPECIFIED
No valid station name or LSN appears after the TO in the command string, or the TO was omitted.

>> ERROR: DESTINATION STATION NOT CONTROLLED BY <SMCS program id>
The station specified in the command is not under the control of the SMCS program.

>> ERROR: DETACH CURRENTLY IN PROGRESS ON FILE <remote file number>, PLEASE WAIT A BIT OR CLEAR THE FILE
Another station is detaching from the specified remote file.

>> ERROR: ENDING QUOTE MISSING
A quoted token was entered at a station that was missing the ending quotation mark character (").

>> ERROR: FILE '<program name>' NOT ON SYSTEM
The program name entered with the command string is not in the disk directory.

>> ERROR: FILE "<file name/number>" UNKNOWN
The remote file name or number specified in the STATUS command string is not presently in the remote file table of the SMCS program.

>> ERROR: FILE "" UNKNOWN
A remote file name or number was omitted from the command string entered.

>> ERROR: FILE "TO" NOT ON SYSTEM
The PASS command entered was immediately followed by the keyword TO and not the expected "program name".

>> ERROR: FILE NAME IS REQUIRED BY 'FILE'
The command syntax requires a file name which was omitted from the command string.

>> ERROR: FREQUENCY MUST BE AN INTEGER
The receive or transmit frequency value must be an integer not exceeding the value 255.

>> ERROR: FREQUENCY MUST BE LESS THAN 256
The FREQUENCY value for either receive or transmit must not be greater than the maximum value 255.

>> ERROR: INTEGER REQUIRED FOR ATTRIBUTE
A line printer size was not specified in the HARDWARE command.

>> ERROR: INVALID CHANGE FIELD "<option>"
The change option entered did not match any valid options.

>> ERROR: INVALID CHANGE FIELD ","
The CHANGE command does not allow a station list.

>> ERROR: INVALID COMMAND, ENTER "HELP" IF NEEDED
The command entered is not a valid command.

>> ERROR: INVALID COMMAND OPTION KEYWORD: "<X..X>"
An invalid or missing keyword option was found in the MAKE command string.

>> ERROR: INVALID COMMAND SYNTAX
The command string was incorrectly entered.

>> ERROR: INVALID DATE OR TIME IN STEP MACRO WAIT COMMAND
An invalid value was given for the WAIT FOR or WAIT UNTIL step macro commands.

>> ERROR: INVALID FILE DESIGNATOR: "<file identifier>"
Either the remote file name or remote file number specified was missing or was not found in the program table or file table.

>> ERROR: INVALID FILE NAME ENTERED
The file name as entered was not found in the disk directory. Either there was an incorrect slash character (/) or no ending quotation mark in a quoted string.

>> ERROR: INVALID FILE: "<file name>"
The remote file name specified in the command string was not found in the file table.

>> ERROR: INVALID INTEGER VALUE
The integer entered for the line printer size in the HARDWARE command syntax was not 120 or 132.

>> ERROR: INVALID KEY ON A STEP MACRO GOTO
An invalid record was specified in the step macro GOTO command.

>> ERROR: INVALID LOGON COMMAND
The only options that can be specified in the LOGON command are ADD, CHANGE, DELETE, or LIST.

>> ERROR: INVALID MACRO NAME
The macro procedure name entered is invalid. HEADERS is a reserved name.

>> ERROR: INVALID OPTION KEYWORD "<token>"
For the HARDWARE command, the only valid peripheral option is LP.

>> ERROR: INVALID PROGRAM NAME: <program name>
A program name was entered with the command string that was not found in the system directory.

>> ERROR: INVALID PROGRAM NAME: "<x...x>"
A program name was not included in the command string.

>> ERROR: INVALID PROGRAM: "<program name>"
The program name entered in the command string is not in the current program table.

>> ERROR: INVALID PROGRAM: ""
This message indicates that a program name or job number was omitted from the command string. A valid program identifier must follow the keyword PROGRAM.

>> ERROR: INVALID REMOVE COMMAND
The station identifier specified in the command syntax is invalid.

>> ERROR: INVALID SOURCE STATION: "x...x"
The characters assumed to be the station identifier from which the messages are to be moved are missing or are non-digits or are not found in the station table.

>> ERROR: INVALID STATION "<station identifier>"
The station name or LSN entered is not in the currently active station table.

>> ERROR: INVALID STATION: <station name>
The station name entered was not found in the station table.

>> ERROR: INVALID USERCODE AND/OR PASSWORD ENTERED
The usercode or password entered was not found to be valid in the usercode/password file.

>> ERROR: LOG-ON DEFAULT COMMAND NOT PRESENT
The LOGON command was entered to list, change, or delete the default log-on command, and the user did not currently have a default log-on command.

>> ERROR: LOG-ON DEFAULT COMMANDS NOT ALLOWED
The LOGON command is only allowed if program switch 3 is set at beginning of job or if the MACRO ON command is entered.

>> ERROR: LOG-ON DEFAULT COMMAND PRESENT
The LOGON command was entered to add a default log-on command, and the user already has a default log-on command.

>> ERROR: LOG-ON DEFAULT COMMAND REQUIRED
The LOGON command was entered to add or change the default log-on command, but no default log-on command was specified.

>> ERROR: LSN <number> NOT CONTROLLED BY THIS MCS
The station LSN entered is currently under the control of another MCS.

>> ERROR: MACRO <command> NOT IMPLEMENTED
The macro command, <command>, that was entered is not currently implemented.

>> ERROR: MACRO CAUSES BUFFER TO EXCEED <number> CHARACTERS
The macro invocation caused the buffer to exceed the maximum number of characters.

>> ERROR: MACRO EXCEEDS <number> CHARACTERS
The macro procedure has exceeded <number> CHARACTERS.

>> ERROR: MACRO LOOPING
A macro invocation was called 16 times.

>> ERROR: MACRO NAME REQUIRED
No macro procedure name was entered in the MACRO ADD command.

>> ERROR: MACRO OPERATIONS NOT ALLOWED
Macro operations are not allowed since program switch 3 is not equal to 1.

>> ERROR: MACRO PRESENT
A MACRO ADD command was performed when the <macro name> was already present.

>> ERROR: MACRO UNKNOWN
The macro procedure specified was not found.

>> ERROR: MAXSTATIONS FOR FILE "<file name>"
EXCEEDED BY <number> STATION(S)-ATTACH NOT SENT
This message indicates that the station(s) specified in the ATTACH command string cannot be attached because the maximum number of stations for the remote file would be exceeded.

>> ERROR: MCP REJECTED CONTROL STRING
A control string was spawned through a ZIP command and was rejected by the MCP.

>> ERROR: MESSAGE COUNT CANNOT BE ZERO
A <number> of messages was specified or recognized as zero.

>> ERROR: MISSING "("
>> ERROR: MISSING ")"
The command decoder was expecting a prefix of "(" or a suffix of ")" to the suboption RECEIVE or TRANSMIT in the CHANGE TRAN, ADDRESS, or FREQUENCY option.

>> ERROR: MISSING PARAMETER
A replacement string was specified in a macro invocation without a parameter.

>> ERROR: MISSING PROGRAM NAME
The SIGN ON or ON command was entered without a program-id specified.

>> ERROR: MISSING RIGHT BRACKET
A right bracket (]) was not specified to terminate the macro invocation.

>> ERROR: MISSING STATION DESIGNATOR AND COUNT
When the MOVE command is entered without parameters, this message is displayed.

>> ERROR: MISSING STATION LIST AND TEXT
The command was entered without an option or station list and possibly with no message text.

>> ERROR: MUST ATTACH FROM A CONTROLLER STATION. TRY 'SIGN ON'.
The ATTACH command can only be initiated from a controller station.

>> ERROR: MUST CLEAR FROM CONTROLLER STATION
The CLEAR command was entered from either a remote ODT or workstation. It is valid only when entered at a controller station.

>> ERROR: MUST DETACH FROM CONTROLLER STATION. TRY 'SIGN OFF'.
The DETACH command is valid only when entered at a controller station.

>> ERROR: MUST SIGN OFF <program name> FIRST
User cannot sign on to another program before signing off the present program to which the user is attached.

>> ERROR: NO MACRO DATA
The macro string was not specified in the MACRO ADD or the MACRO CHANGE command.

>> ERROR: NO MACRO NAME
A macro procedure name was not specified in the command.

>> ERROR: NOT A CONTROLLER STATION–CANNOT PERFORM MOVE
The MOVE command cannot be accepted from a workstation or remote ODT.

>> ERROR: NOT A CONTROLLER STATION–CANNOT PERFORM REMOVE
A REMOVE command was entered from either a workstation or remote ODT. It is valid only when entered at a controller station.

>> ERROR: NOT A CONTROLLER; CANNOT PERFORM SIGNAL CHANGE
A signal change can only be accepted from a controller station.

>> ERROR: NOT A VALID PHONE NUMBER
The phone number entered contains characters other than digits and separation hyphens (-) or separation blanks.

>> ERROR: NOT ALLOWED FROM SPO
A macro invocation or command is not allowed from the system ODT.

>> ERROR: NULL STATION LIST IN ATTACH
This means that a station was not specified in the ATTACH command string.

>> ERROR: NULL STATION LIST: NO MESSAGE SENT
A station(s) to which the message text was directed was not found in the station table and is assumed to be invalid.

>> ERROR: ON OR OFF EXPECTED
This message appears when a directive code is not supplied with the CHANGE command when any of the following change keywords is entered: ENABLED, READY, DIAGNOSTIC, LOGICALACK, or GOODRESULTS.

>> ERROR: ONLY 10 REPLACE OPERATIONS ALLOWED
Only 10 parameters and their replacement strings are allowed in a macro invocation.

>> ERROR: PASS REQUIRES '<STRING>' FOLLOWING PROGRAM NAME.
Message text was omitted from the command string.

>> ERROR: PHONE NUMBER TOO LARGE
The phone number entered exceeds 20 digits.

>> ERROR: PROGRAM "<program identifier>" UNKNOWN
The program name or job number specified in the STATUS command string is not in the program table.

>> ERROR: PROGRAM "<program name>" NOT EXECUTABLE
The program name entered is not that of an executable program code file.

>> ERROR: PROGRAM "" UNKNOWN
A valid program identifier (name or job number) was omitted from the command string entered.

>> ERROR: PROGRAM SPECIFIED IS CURRENTLY RUNNING
The program specified in the INITIATE command is already executing.

>> ERROR: RECALL TABLE FULL-WAIT A BIT AND REENTER COMMAND
The recall table does not contain an entry to store the move information as requested.

>> ERROR: RECEIVE OR TRANSMIT EXPECTED
The command decoder expected either a RECEIVE or TRANSMIT suboption code for a TRAN, ADDRESS, or FREQUENCY change.

>> ERROR: REPLACE OPERATIONS CAUSE
MACRO TO EXCEED <number> CHARACTERS
The control string resulting from the invocation of the macro procedure exceeded the maximum length (<number> characters).

>> ERROR: RETRIES MUST BE AN INTEGER
To change the station attribute "retries", the command syntax requires a positive integer with a value less than 256.

>> ERROR: RETRIES MUST BE LESS THAN 256
To change the station attribute "retries", the command syntax requires a positive integer with a value less than 256.

>> ERROR: SOURCE STATION NOT CONTROLLED BY <SMCS program id>
The station specified is attached to another MCS.

>> ERROR: STATION ACTIVE; CANNOT CHANGE SIGNAL CHARACTER
The signal character for SMCS cannot be changed with the SIGNAL command if there are stations active.

>> ERROR: STATION <station name> CANNOT BE MADE "READY"
A READY command was entered for a station which is under the control of another MCS.

>> ERROR: STATION <station name> NOT CONTROLLED BY THIS <SMCS program id>
The station name or LSN entered with the command string is presently marked as active under another MCS.

>> ERROR: STATION "<station identifier>" UNKNOWN
The station name or LSN entered cannot be found in the station table.

>> ERROR: STATION "<token>" UNKNOWN
The station name or LSN entered cannot be found in the station table.

>> ERROR: STATION "" UNKNOWN
The CHANGE command was entered to the SMCS without a station identifier or station identifier list.

>> ERROR: STATION ALREADY SIGNED OFF
A sign-off is already in progress for the station.

>> ERROR: STATION NOT CONTROLLER; CANNOT 'MAKE' CHANGE
A CHANGE command is only valid from a controller station.

>> ERROR: STATION NOT SIGNED ON
A station that is not presently signed on to the SMCS cannot be signed off.

>> ERROR: STATION(S) <station name list> CURRENTLY ACTIVE
Attachment of the displayed station(s) cannot be granted as they are currently attached to another remote file.

>> ERROR: STEP MACRO LOOPING
The step macro procedure currently executing is looping.

>> ERROR: STRING REQUIRED FOR MCP COMMUNICATION.
The MCP command string requires an attribute string preceded by a question mark (?).

>> ERROR: SUPERFLUOUS TEXT AFTER FILE NAME
The user entered additional data in a command string following the file name.

>> ERROR: SYNTAX REQUIRES 'TO' BEFORE DESTINATION
The command syntax requires the keyword TO separating the source station identifier and the destination station identifer.

>> ERROR: THE MAIL FILE IS TEMPORARILY LOCKED
The SMCS is in the process of establishing a new MAIL file.

>> ERROR: THIS STATION ALREADY ATTACHED TO "SMCS"
When a SIGN ON command is entered with a program name equal to the SMCS program name, it is rejected and this message is returned.

>> ERROR: THIS STATION IS CURRENTLY IN USE BY '<usercode>'.
An attempt was made to log on a station that another user is currently logged on to.

>> ERROR: THIS STATION IS NOT CURRENTLY LOGGED ON.
Cannot log off a station that is not currently logged on.

>> ERROR: VALID <IDENTIFIER> REQUIRED
The ID command has been entered without the required identifier character string.

>> ERROR: VALID <USERCODE>/<PASSWORD> REQUIRED
The USER command was entered without the required usercode/password combination.

>> ERROR: ZIP TYPE REQUIRES USER TO BE LOGGED ON
The user must be logged on with a usercode/password.

>> WARNING: DETACH RECEIVED FOR RF# <n>: NO STATIONS
A DETACH message was received from the network controller which did not contain a station list.

>> WARNING: DETACH.REPLY RECEIVED FOR RF# <n> : UNKNOWN
A DETACH reply was received from the network controller for a remote file that is not known to the SMCS program.

>>> WARNING: "FILE TABLE" FULL
An attempt was made to assign an additional remote file, but in searching the file table an available entry could not be found. The requesting station was informed of this condition.

>>> WARNING: INVALID MESSAGE TYPE "<nn>" ON REMOTE READ
The occurrence of this message indicates that an invalid message (type) was read from the remote file queue. It has been logged in the TRACE file if the disk trace or printer trace option is on. This invalid message appears in the TRACE file just prior to this warning message.

>>> WARNING: JOB "<program id>" EXECUTED WITHOUT ATTRIBUTES
The corresponding entry in the JOBS file for the program has an MCP zip string specified which contains characters in excess of the maximum SMCS buffer size (1200). The zip string has been truncated to the last semicolon (up to a maximum of 1200 characters).

>>> WARNING: JOB "<program id>" FIRED-UP WITHOUT ATTRIBUTES
The corresponding entry in the JOBS file for the program has an MCP zip string specified which contains characters in excess of the maximum SMCS buffer size (1200). The zip string has been truncated to the last semicolon (up to a maximum of 1200 characters).

>>> WARNING: JOB "<program name>" CANNOT BE FIRED-UP
The job could not be spawned by the SMCS program for one of several reasons: (1) there is no available entry in the program table, (2) there is no available entry in the file table, (3) the program was either not on disk or was locked, or (4) the program specified was not a code file.

>>> WARNING: MCP REJECTED JOB EXECUTE FOR "<program id>"
The AUTO-START option in the JOBS file was specified for the program and the MCP rejected the zip string spawned by the SMCS program.

>>> WARNING: UNKNOWN LSN (<nnn>) RECEIVED; MESSAGE REJECTED
This message means that a message was received in the remote file which contained an invalid station LSN. The message received was logged in the TRACE file.

>>>> OPERATOR ERROR: THIS MCS IS NOT
INTENDED FOR OPERATION UNDER A SESSION NUMBER
The SMCS cannot be executed under a session number.

>>>> OPERATOR ERROR: THIS MCS IS NOT
INTENDED FOR OPERATION UNDER A USERCODE
The SMCS cannot be executed under a usercode.

# APPENDIX A
# GLOSSARY OF TERMS

**active jobs**
Programs that are currently executing in the mix of the system.

**active schedule**
Contains those jobs that have satisfied all the requirements for execution and are only waiting for memory space to run.

**active station**
A workstation that is currently eligible for entering or accepting messages.

**application program**
A program that processes data and is unique to one type of application. It is also referred to as a user program.

**beginning of job (BOJ)**
The execution of a single program unit to be performed by the system.

**BOJ**
A mnemonic abbreviation for beginning of job.

**broadcast**
The simultaneous sending of information to a number of terminals.

**buffer**
Temporary storage that is used during the transfer of data from one device to another.

**character**
A digit, letter, or special character.

**communication line**
Any medium, such as a wire or a telephone circuit, that connects a terminal or computer to a computer.

**controller station**
A workstation with an extended capability that allows restricted SMCS commands to be entered and processed, altering the network configuration and/or system control attributes.

**Controlling Remote File (CRF)**
The file with headers to which a station was most recently assigned, either with an ATTACH or with an OPEN. If the primary file has headers, it is the CRF; otherwise, the secondary file is the CRF.

**conversational mode**
An operational mode in which message data (full or partial messages) is used instead of, or in addition to, control characters as replies for message information.

**CRF**
A mnemonic abbreviation for controlling remote file.

**data**
Data is any representation of information, such as characters, to which meaning can be assigned.

**data comm**
The acronym for data communications.

**data communications**
Data communication is a method of transferring encoded data by means of electrical transmission between two or more terminals and/or systems.

**DIRECTORY file**
A disk data file containing the macro procedures added or changed by the MACRO command.

**disk directory**
A disk-resident table that contains the name and type of file, together with a pointer to the disk file header or subdirectory for all permanent files which reside on disk.

**disk pack**
A secondary data storage device that can be moved on-line or off-line.

**end-of-file (EOF)**
Signals that the last record of a file has been read.

**end of job (EOJ)**
Signals that a job has been completed.

**EOF message**
A special interface message sent to an application program as a go to end of job directive.

**EOJ**
A mnemonic abbreviation for end of job.

**family name**
A disk pack identifier.

**file-id**
The file-id is used to identify the file identifier of a file.

**file name**
All file names used on the system must be unique to prevent duplication of file names. The file name incorporates allowable combinations of the family name, multi-file identifier, file identifier, and slash character (/) separators. Refer to the B 1000 System Software Operation Guide, Volume 1, for more information on the file names and file naming conventions.

**file security**
The procedures or special devices used to prevent access to or use of data or programs without authorization.

**HASP**
A program product designed to perform the functions normally associated with an off-line support computer. HASP is a method of automatic spooling and priority scheduling for an unlimited number of remote peripherals. It is normally in a mode of receiving data from larger systems over data communication lines.

**identifier**
A word consisting of from one to ten alphabetic, numeric, or special characters in combination.

**INFO file**
A disk file containing the current system news message and the current global signal character.

**integer**
A whole number as distinguished from a fraction.

**interprogram communication**
A reference to the functional process of two or more jobs (programs) sending data messages back and forth between themselves. This process utilizes the MCP queue mechanism.

**I/O**
A mnemonic abbreviation for input/output.

**job**
An application program or system software element (such as SMCS) that performs a job (or task). It is generally composed of many job steps and usually performs many functions.

**JOBS file**
A disk data file containing system software (such as CANDE, RD, and SYCOM) and application program name entries for all programs to be executed in conjunction with the SMCS. Each entry includes any special requirements (MCP parameters) about the manner in which the program must be executed.

**job number**
A unique 4-digit number assigned by the operating system (MCP) to each job that is executed under its control.

**job spawning**
Job spawning is the ability to programmatically execute jobs and maintain an adequate level of control over these jobs during their execution.

**keyboard**
A device for the encoding of data by key depression, which causes the generation of the selected code element (character).

**keyword**
A symbol that identifies a parameter.

**keyword parameter**
A parameter that consists of a keyword followed by one or more values.

**line adapter**
A line adapter consists of a logic card used to interface a data set or communications line into a multiline or single-line I/O control.

**logical station number (LSN)**
The identification number assigned by the network controller to a given station. The logical station number has no relationship to file names, but is determined by the order in which a station is defined in the network controller.

**LSN**
A mnemonic abbreviation for logical station number.

**macro procedure**
A macro procedure is a MCP control string or SMCS command which is stored in the DIRECTORY file. The macro procedure is accessible by invoking the macro name. Replacement parameters can also be passed to the macro procedure enabling the creation of general purpose macro procedures.

**MAIL file**
A disk data file in which messages from a program, the MCP, or terminal are stored. The system of keeping these messages is defined as the MAIL function of the SMCS.

**Master Control Program (MCP)**
The Master Control Program is the operating system used to control the B 1000 systems.

**MCP**
A mnemonic abbreviation for Master Control Program.

**MCS**
A mnemonic abbreviation for Message Control System.

**message**
A sequence of characters arranged in a form suitable for the purpose of conveying information. A message contains the data to be conveyed (the text) and, in addition, can contain communication characters to aid in the routing or handling of the message.

**Message Control System (MCS)**
A program that controls message flow between application programs and the network controller.

**message header**
Every message moved between the network controller and the SMCS is prefixed with a message header. This message header contains station attributes necessary to identify a station.

**MLC**
A mnemonic abbreviation for multiline control.

**multifile identifier**
An identifier which is a file name or the name given to identify a main directory file name with subdirectory entries.

**multiline control (MLC)**
The MLC is an I/O control device that provides the functional control between computer hardware and the line adapters. The control is an integral part of a multiline communications subsystem.

**NDL**
A mnemonic abbreviation for the Network Definition Language compiler.

**network**
A number of data communication lines connecting the local system with remote stations (terminals).

**network controller**
The NDL-generated data communication program used by the SMCS to interface with the stations (terminals).

**Network Definition Language (NDL)**
A descriptive, free-form language for defining and implementing a data communications network. The NDL compiler analyzes the input statements and generates a custom network controller.

**network reconfiguration commands**
Commands which change certain station attributes or otherwise affect network controller operation. They are only permitted from a controller station.

**ODT**
A mnemonic abbreviation for Operator Display Terminal.

**off-line**
This term is used to describe terminal equipment which is not not connected to a transmission line. It can also describe other devices not in direct communication with the central processing unit.

**on-line**

Used to describe terminal equipment which is connected to a transmission line. It can also describe other devices in direct communication with the central processing unit.

**PASS command**
A station command which directs the SMCS to send a message to a specified program, without the station being signed on to that program.

**password**
A unique string of characters that a program, computer operator, or user must supply to meet security requirements before gaining access to files.

**program-PASS message**
A message, written to an application program by the SMCS or written to the SMCS from an application program, which allows an operator at a station to request a function of a program in the local system without being signed on to the program.

**program table**
Used by the SMCS program to store information about the programs spawned as a result of a SIGN ON or a PASS and those initiated at a station which opens a remote file.

**queue**
A logical collection of messages awaiting transmission or processing.

**relative station number (RSN)**
The number by which a user program with a remote file using a remote key uniquely identifies a station. An RSN equal to zero implies a control message. An RSN equal to one implies the first station in the file FAMILY statement in the network controller. RSNs proceed sequentially through the stations declared in the FAMILY statement, except that a Controlling Remote File can modify the LSN list of the file and thereby modify the RSNs of the remote file. If a station is detached from a file with a remote key, the RSNs remain unchanged. If a station is attached to a file with a remote key, the new RSN is the same as the old RSN if the station was attached previously; otherwise, the RSN is one larger than the greatest RSN previously associated with the file.

**remote file**
A file declared in a program which, in conjunction with the network controller, provides input, output, or input/output with a set of data communication devices.

**Remote Job Entry (RJE)**
The Remote Job Entry program is a method by which a remote computer executes or compiles programs sent to it by other computers which are considered terminal devices.

**remote ODT**
A station that is allowed to control jobs on a local system by entry of certain restricted MCP commands similar to those entered at the system ODT.

**restricted commands**
Those commands which are designated as ODT commands and are restricted by the SMCS to remote ODT stations only.

**RJE**
A mnemonic abbreviation for remote job entry.

**RSN**
A mnemonic abbreviation for relative station number.

**semantics**
The relationship between symbols and their meanings.

**single-line control (SLC)**
This is an I/O control device that provides the functional control between the mainframe of a computer and a line adapter. This control is an integral part of a single-line communications subsystem.

**SLC**
A mnemonic abbreviation for single-line control.

**SMCS**
A mnemonic abbreviation for the Supervisory Message Control System.

**spawning**
See job spawning.

**station**
Same as terminal.

**station identifier**
Station identifier is the logical station number or station name assigned in the network controller.

**system disk**
A disk pack or disk cartridge that is initialized as a system disk. A system disk is under control of the MCP and one or more must be present on the system for the MCP to function.

**system ODT**
The operator display terminal on the local system.

**system software**
System software programs reference basic utility programs required by the system to operate.

**system software element**
A specific program from a system software package, such as DMPALL or SMCS.

**task**
A routine or machine run. A program can be broken into several tasks to more effectively utilize the B 1000 computer's processing time.

**terminal**
This is an input/output device designed to both receive and send source data in an environment associated with the job to be performed. Also, it is capable of sending entries to, and obtaining output from, the system of which it is a part.

**unrestricted commands**
Any SMCS or MCP command that is allowed from any station.

**user commands**
A command which accepts and validates a usercode/password and permits that user to be logged on to the SMCS.

**user program**
This is a program that processes data and is unique to one type of application. Also referred to as an application program.

**usercode**
A symbol which consists of from one to eight characters to identify a user.

**usercode index**
The relative record number of the usercode/password pair in the (SYSTEM)/USERCODE file.

**VDT**
A mnemonic abbreviation for video display terminal.

**video display terminal (VDT)**
A video type workstation (terminal) such as the MT 985, MT 983, TD 830, and SR 100 terminal devices.

**waiting schedule**
Contains the names and run-time attributes of jobs that are waiting to be placed into the active schedule.

**workstation**
A terminal which does not have any SMCS control capabilities such as a remote ODT or controller station.

**ZIP execute**
An execute statement sent from a program to the MCP requesting that a job be scheduled and run.

# APPENDIX B
# CHARACTER SET

The following characters define the SMCS character set. They are arranged by columns in collating sequence with "blank" low and the number 9 high.

| | | | | | |
|---|---|---|---|---|---|
| blank | : | s | I | S | 3 |
| ( | = | t | J | T | 4 |
| * | " | A | K | U | 5 |
| ) | a | B | L | V | 6 |
| ; | e | C | M | W | 7 |
| – | h | D | N | X | 8 |
| / | l | E | O | Y | 9 |
| , | n | F | P | Z | |
| > | p | G | Q | 1 | |
| ? | r | H | R | 2 | |

Eight other characters are acceptable to the SMCS program only for use as signal characters:

[ & $ % ] # @ +

# APPENDIX C
# RESTRICTED MCP COMMANDS

Security restrictions impose certain constraints on MCP commands from remote work stations. Also, entry of certain commands such as those which alter system options, purge, clear, or assign system peripherals, is disallowed from a remote work station. A list of the MCP command subset that is permitted from work station terminals follows. The MCP command subset is listed alphabetically as described in the B 1000 Systems System Software Operation Guide, Volume 1.

Mix-related MCP commands can refer only to jobs initiated by the same remote terminal where the initial command was entered. All MCP commands must be entered in upper-case letters. Any command in the following list which is followed by (US) must be entered with a valid usercode/password combination.

Library Maintenance Instructions

ADD (US)
CH Change (US)
COPY (US)
RE Remove (US)

Program Control Instructions

| | |
|---|---|
| CO | Compile (US) |
| DY | Dynamic |
| EX | Execute (US) |
| MH | Modify Header (US) |
| MO | Modify (US) |

Program Control Instruction Attributes

| | |
|---|---|
| AF | After |
| AN | After. Number |
| CA | Conditional |
| CG | Charge (certain restrictions apply) |
| DS | Dynamic.Spaces |
| FI | File |
| FR | Freeze |
| HO | Hold |
| ID | Intrinsic.Directory |
| IN | Interpreter |
| IT | Intrinsic.Name |
| ME | Memory |
| OV | Override |
| PR | Priority |
| SC | Schedule.Priority |
| SW | Switch |
| TH | Then |
| UC | Unconditional |
| UF | Unfreeze |
| UV | Unoverride |
| VI | Virtual.Disk |

System Control Instructions

| | |
|---|---|
| AX | Response to ACCEPT |
| BF | Display BACKUP files (US) |
| CP | Compute |
| CU | Core usage |
| DF | Date of file (US) |
| DM | Dump memory and continue |
| DP | Dump memory and discontinue |
| DS | Discontinue program |
| FN | Display internal file name |
| FR | Final reel of unlabeled tape file |
| FS | Force from schedule |
| GO | Resume stopped program |
| HS | Hold in waiting schedule |
| HW | Hold in waiting schedule until job EOJ |
| MR | Close output file with purge |
| MX | Display MIX |
| OF | Optional file response |
| OK | Continue processing |
| OL | Display peripheral status |
| PD | Display directory |
| PM | Print memory dump |
| PR | Change priority |
| PS | Prod schedule |
| QF | Query file (US) |
| QP | Query program |
| RB | Remove backup files (US) |
| RM | Remove duplicate disk file |
| RS | Remove jobs from schedule |
| SP | Change schedule priority |
| ST | Suspend processing |
| SW | Set switch |
| TD | Time and date |
| TI | Time interrogation |
| TS | Test switches |
| UL | Assign unlabeled file |
| WD | Display MCP date |
| WM | Display current MCP and interpreter |
| WS | Display schedule |
| WT | Display MCP time |
| WY | Program status interrogation |

# APPENDIX D
# SUMMARY OF FILES

## SMCS FILE INFORMATION

Table D-1 lists all the files used by the SMCS program along with their internal and external file names, device assignment, I/O use, recordsize (in bytes), and the number of buffers.

**Table D-1. Summary of Files**

| File Number | Internal Name | External Name | Device | Use | Record Size | Buffers |
|---|---|---|---|---|---|---|
| 0 | REMOTE | MCSREMOTE | remote | I/O | 1330 | 5 |
| 1 | INIT.QUEUE | SMCS/INIT | queue | O | 1330 | 2 |
| 2 | TRACE.FILE | SMCS/TRACE | PBD | O | * | 2 |
| 3 | MCPQ | SMCS/MCPQ | queue | I | 1330 | 2 |
| 4 | SCHQ | SMCS/SCHQ | queue | I | 1330 | 2 |
| 5 | JOBQ | SMCS/JOBQ | queue | I | 1330 | 2 |
| 6 | MSGQ | SMCS/MSGQ | queue | I | 1330 | 2 |
| 7 | MACROFILE | MACROFILE | disk | I/O | * | 2 |
| 8 | US.DIRECTO | SMCS/DIRECTORY | disk | I/O | 1330 | 1 |
| 9 | MAIL | MAIL/<day> | disk | O | 80 | 1 |
| 10 | NEW.JOBS | SMCS/JOBS | disk | I/O | 90 | 1 |
| 11 | JOBS | SMCS/JOBS | disk | I | * | 1 |
| 12 | GROUPS | SMCS/GROUPS | disk | I/O | 180 | 1 |
| 13 | INFO | SMCS/INFO | disk | I/O | 180 | 1 |
| 14 | LIST | SMCS/LIST | disk | I | 80 | 1 |

NOTES

1. An asterisk character (*) in the record size column indicates that the record size of the file is used.

2. All external file names that have a family identifier of SMCS are changed to the family identifier of the SMCS program if it differs from SMCS.

# APPENDIX E
# THE MAIL FILE

The MAIL file is a disk file named MAIL/<day> maintained by the SMCS program. <day> is the name of the current day. The MAIL file contains entries made by the SMCS program whenever a Program-PASS message command or a MAIL message command is entered from a terminal or there are system ODT output messages for a user. The MAIL file for the day is created when the SMCS program is first executed. If the SMCS program is executed and a MAIL file exists for that day, it is opened and spaced to the next available record where entries are made. At midnight, the SMCS automatically closes and locks the current MAIL file and creates a new MAIL file for the next day. The format of the MAIL file record is as follows:

| Field Name | Length in Bytes | Description |
|---|---|---|
| Identifier | 10 | Message identifier as specified in Program-PASS input message, MAIL message, or ODT output message. This identifier can be a usercode, station name, or the first 10 characters of a program name. |
| filler | 1 | Not used. |
| Time of Day | 5 | The time of entry for this message into the MAIL file. The format is HH:MM where HH is a 2-character military hour and MM is a 2-character minute value. |
| filler | 1 | Not used. |
| Message | 63 | Message text, left justified. If a message is longer than 63 bytes, the remainder of the text is written in the immediately following record with identical identifier and time values. |

# APPENDIX F
# THE TRACE FILE

A disk file is employed by the SMCS program to provide a trace of input-output activity. The file is maintained in two levels: previous, which is labeled SMCS/TRACER, and current, which is labeled SMCS/TRACE.

The current trace file, SMCS/TRACE, maintains the last 495 trace lines. When the end-of-file record is reached, the SMCS program resets the record pointer to 1 and starts writing the trace records from the beginning of the file. The time and date are prefixed to each trace line written into the file, making it easy to determine the sequence of events. The record size of the disk trace record is 120 bytes and can be printed using the DMPALL program with the following syntax:

```
EX DMPALL
<job number of DMPALL>AX LIST SMCS/TRACE A
```

The file can also be printed using the LIST option of the TRACE command. Refer to section 5, SMCS Operating Instructions, for an explanation of the TRACE command.

## RECORD FORMAT

The trace record consists of the trace record type, followed by the system time and date, followed by the trace text. If the trace text is too long to be written in one record, the remainder of the text is written in the immediately following record. The format of the trace text depends on the value of the trace record type. A description of the trace record types follows:

### IQUE SAVE

If text follows the command to log on to the SMCS program, it is read and this action is recorded in the IQUE SAVE trace record. After the log on is complete, the SMCS program processes the text.

### JOBQ IN

The trace text is the message read by the SMCS program from the control queue JOBQ. This control queue is used by SMCS to receive messages from the MCP regarding jobs spawned by the SMCS as a result of receiving the first SIGN ON or PASS for a program.

### MCPQ IN

The trace text is the message read by the SMCS program from the control queue MCPQ. This control queue contains all control messages, input and output, regarding a secured control string spawned by the SMCS program. The format of messages received in a control queue is described in the B 1000 Systems System Software Operation Guide, Volume 1.

## MSGQ CHIN

When two or more stations are waiting for a file open and the SMCS program receives the file open, the SMCS program determines what stations are waiting for the open. This action is recorded in the MSGQ CHIN trace record. If not all of the waiting stations can be approved in the open, the SMCS program changes the file number of the waiting stations which cannot be included in the first file open. The SMCS program then spawns another copy of the program. The remote file number change is recorded in the MSGIN CHOT trace record. The text of the two trace records, MSGQ CHIN and MSGQ CHOT, consists of an S or P in the first byte indicating whether it is for a PASS or SIGN ON command, followed by a 3-byte station index, a 3-byte remote file number, and any text entered following the SIGN ON <program-name> or PASS <program-name> command.

## MSGQ IN

If the LOG-ON option is specified in the JOBS file for a program, and after the file open or station attach reply message is sent to the network controller by the SMCS program, the SMCS program reads the usercode/password pair of the station which initiated the PASS or SIGN ON to send it as first input to the program. This action is recorded in the MSGQ IN trace record.

## MSGQ SAVE

If the LOG-ON option is specified in the JOBS file for a program, and a PASS or SIGN ON command is received for that program, the SMCS program saves the usercode/password of the user initiating the SIGN ON or PASS and records this action in MSGQ SAVE.

## REMOTE IN

The trace text is the message received on a remote file read by the SMCS program, which includes all messages from terminals in the SMCS system and from the network controller.

## REMOTE OUT

The trace text is the message written to the SMCS remote file by the SMCS program, which includes all messages to terminals in the SMCS system and to the network controller. The message is prefixed with a header. The format of the message depends on the message type in the header.

## RF INFO-2

The trace text is the reply from the network controller to a request by the SMCS program for initial terminal status.

## SCHQ FIRE

The trace text is a message read by the SMCS program from the zip queue SCHQ. The SCHQ FIRE record type records that the queue is being read to get the MCP response to the spawning of a program at beginning of job (BOJ) of the SMCS program as a result of the AUTO-START option in the JOBS file being set for that program. The format of messages received in a zip queue is described in the B 1000 Systems System Software Operation Guide, Volume 1.

## SCHQ IN

The trace text is a message read by the SMCS program from the zip queue SCHQ. This zip queue contains schedule and DATA card messages for jobs spawned by the SMCS program as a result of receiving the first SIGN ON or PASS for the program. The format of messages in a zip queue is described in the B 1000 Systems System Software Operation Guide, Volume 1.

## ODT OUT

The trace text consists of the output message displayed on the ODT by the SMCS program.

## ZIP

The trace text consists of the message spawned through a ZIP command by the SMCS program.

# APPENDIX G
# SMCS PROGRAM-PASS MESSAGES

## GENERAL

The purpose of the PASS command in the SMCS program and the program-pass message interface is to allow an operator at a terminal to request a function of a program in the local system without being signed on to the program. The program must have opened a remote file which was approved by the SMCS program.

## PROGRAM-PASS OUTPUT MESSAGES

The program-pass output message is written by the SMCS program. The purpose of the program-pass output message is to transfer a message from a station under the control of the SMCS to a program. The program-pass output message consists of two parts:

1. A message header for remote files with headers or a key for remote files with remote key.
2. The message text, which has a maximum size of 1920 bytes. The format of the message text is determined by the PROTOCOL file attribute as specified for the remote file of the program. The PROTOCOL file attribute can be set or changed using the FILE program control attribute instruction. Refer to the B 1000 Systems System Software Operation Guide, Volume 1, for a complete description of the FILE statement.

   If the value for the PROTOCOL attribute is 0, the message text consists only of the message to be passed and the message type is 01. A value of 0 for the PROTOCOL attribute is the default value.

   If the value for the PROTOCOL attribute is 1, the message text consists of the usercode index of the usercode logged on at the station that passed the message. The rest of the message consists of the actual message to be passed. The purpose of a PROTOCOL attribute equal to 1 is to provide information in the pass message about the station which passed the message.

   If the value for the PROTOCOL attribute is 2, the message text consists of the parent queue name of the control queue used by the SMCS program to spawn control string initiated from that station, the usercode and password, session number, and signal character of the station. The rest of the message text consists of the message to be passed. This information enables the receiving program to spawn jobs on behalf of the station sending the message.

   If the value for the PROTOCOL attribute is 3, the message text consists only of the message to be passed. The purpose of Protocol Type 3 is to allow a program-pass data message to be passed to a program. Since no other information is received with the message, the receiving program sets the message type to 00 to send any reply message directly to the station which passed the message as indicated by the Logical Station Number (LSN) in the message header or the Relative Station Number (RSN) in the remote key.

## Message Header Format

The format of the message header is given in table G-1.

**Table G-1. Program-Pass Message Header Format**

| Field Name | Field Length |
|---|---|
| MESSAGE_TYPE | 99 |
| FILLER | X |
| LSN | 999 |
| MESSAGE_TEXT_SIZE | 9999 |
| REMOTE_FILE_NUMBER | 999 |
| FILLER | X(37) |

Semantics:

MESSAGE_TYPE
The MESSAGE_TYPE field is set to 61 if the PROTOCOL attribute of the remote file of the program is 1, 2 or 3. If the PROTOCOL attribute is 0, the program sets the MESSAGE_TYPE field to 01.

LSN
The LSN field is the Logical Station Number (LSN) of the station to which the message is to be written.

MESSAGE_TEXT_SIZE
The MESSAGE_TEXT_SIZE field is the size of the message that is to be written.

REMOTE_FILE_NUMBER
The REMOTE_FILE_NUMBER field is the remote file number of the program for which the message is being written.

## Remote Key Format

The remote key format is given in table G-2.

**Table G-2. Program-Pass Remote Key Format**

| Field Name | Field Length |
|---|---|
| RSN | 999 |
| MESSAGE_TEXT_SIZE | 9999 |
| MESSAGE_TYPE | 999 |

Semantics:

RSN
The RSN field is the Relative Station Number (RSN) of the station to which the message is to be written.

MESSAGE_TEXT_SIZE
The MESSAGE_TEXT_SIZE field is the size of the message that is to be written.

MESSAGE_TYPE
The MESSAGE_TYPE field is set to 61 if the PROTOCOL attribute of the remote file of the program is 1, 2 or 3. If the PROTOCOL attribute is 0, the program sets the MESSAGE_TYPE field to 01.

## Text Format

The format of the message text is determined by the PROTOCOL attribute as specified for the remote file of the program. The PROTOCOL file attribute can be set or changed using the FILE program control attribute instruction. Refer to the B 1000 Systems System Software Operation Guide, Volume 1, for a complete description of the FILE statement.

The format of the text is dependent upon the PROTOCOL attribute of the remote file as defined in table G-3.

**Table G-3. Program-Pass Output Message Text Format**

| PROTOCOL Type | Field | Field Length |
|---|---|---|
| 0 | TEXT | X(MESSAGE_TEXT_SIZE) |
| 1 | USERCODE_INDEX | 9999 |
| | FILLER | X(10) |
| | SUB_TEXT | X(MESSAGE_TEXT_SIZE - 14) |
| 2 | PARENT_QUEUE_NAME | X(20) |
| | USERCODE | X(10) |
| | PASSWORD | X(10) |
| | SESSION_NUMBER | 9999 |
| | SIGNAL_CHARACTER | X |
| | FILLER | X |
| | SUB_TEXT | X(MESSAGE_TEXT_SIZE - 54) |
| 3 | TEXT | X(MESSAGE_TEXT_SIZE) |

Semantics:

**Protocol Type 0**

TEXT
For a PROTOCOL attribute equal to 0, only the TEXT field of the message text is to be passed.

**Protocol Type 1**

USERCODE_INDEX
The USERCODE_INDEX field is the usercode index of the usercode and password that is logged on at the station.

SUB_TEXT
The SUB_TEXT field is the actual message to be passed.

**Protocol Type 2**

PARENT_QUEUE_NAME
The PARENT_QUEUE_NAME field is the name of the control queue used by the SMCS program to spawn MCP control strings initiated from the station.

USERCODE
The USERCODE field is the usercode that is logged on at the station.

PASSWORD
The PASSWORD field is the password that is logged on at the station.

SESSION_NUMBER
The SESSION_NUMBER field is the Logical Station Number (LSN) of the station. (The LSN is used as the session number in the spawning of jobs by the SMCS program.)

SIGNAL_CHARACTER
The SIGNAL_CHARACTER field is the signal character of the station.

SUB_TEXT
The SUB_TEXT field is the text of the message that is to be passed.

**Protocol Type 3**

TEXT
The TEXT field is the message text that is to be passed.

# PROGRAM-PASS INPUT MESSAGES

The program-pass input message is written by the application program in response to a message passed to it and is read by the SMCS program. The format of the message consists of two parts:

1. A message header for remote files with headers, or a key for remote files with remote keys.
2. The message text, which has a maximum length of 1920 bytes, is subdivided into a 43-byte sub-header followed by the actual text of the message. This sub-header consists of a sub-type field, a sub-variant, and a destination field. The sub-type field determines the format of the destination field. Either a filler, the usercode index, remote file number, usercode, program name or user identifier is the destination field depending on the value of the sub-type field. The sub-variant field determines whether or not the message is to be put in the MAIL file.

## Message Header Format

The message header for program-pass input messages is identical to the message header for program-pass output messages except that the MESSAGE_TYPE field is equal to 60.

Refer to table G-1 for the program-pass remote key format.

## Remote Key Format

The remote key format for program-pass input messages is identical to the remote key format for program-pass output messages except that the MESSAGE_TYPE field is equal to 60.

Refer to table G-2 for the program-pass message header.

## Text Format

The message text format is given in table G-4.

**Table G-4. Program-Pass Output Text Format**

| Field Name | Field Length |
|---|---|
| SUB_TYPE | 99 |
| SUB_VARIANT | 9 |
| DESTINATION | X(30) |
| FILLER | X(10) |
| SUB_TEXT | X(MESSAGE_TEXT_SIZE - 43) |

Semantics:

SUB_TYPE
The SUB_TYPE field determines the formation of the DESTINATION field.

DESTINATION
The DESTINATION field contains the information used in determining the destination of the message. The format of the DESTINATION field varies depending on the value of the SUB_TYPE field as follows:

| SUB_TYPE | DESTINATION | Field Length |
|---|---|---|
| 0 | FILLER | X(30) |
| 1 | USERCODE_INDEX | 9999 |
| | FILLER | X(26) |
| 2 | REMOTE_FILE_NUMBER | 999 |
| | FILLER | X(27) |
| 3 | USERCODE | X(10) |
| | FILLER | X(20) |
| 4 | PROGRAM_NAME | X(30) |
| 5 | USER_IDENTIFIER | X(10) |
| | FILLER | X(20) |

SUB_TYPE = 0

The DESTINATION field consists of a 30-byte filler. The destination of the message is the Logical Station Number (LSN) if the message header format is used, or the Relative Station Number (RSN) if the remote key format is used. The MAIL identifier used is the usercode or identifier of the station. If the destination cannot be found, the message is not put in the MAIL file or sent to any station.

SUB_TYPE = 1

The DESTINATION consists of the USERCODE_INDEX field and filler. The destination of the message is each station logged on with the USER command whose usercode index matches the USERCODE_INDEX field. The MAIL identifier is the usercode associated with the USERCODE_INDEX field.

SUB_TYPE = 2

The DESTINATION field consists of the REMOTE_FILE_NUMBER field and filler. The destination of the message is each station attached to the remote file number specified in the REMOTE_FILE_NUMBER field. The MAIL identifier is the usercode or identifier of the station(s) attached to the remote file. If the destination station(s) has no usercode or identifier, then the message is put in the MAIL file under the program name associated with the remote file whose number matches the REMOTE_FILE_NUMBER field.

SUB_TYPE = 3

The DESTINATION field consists of the USERCODE field and a filler. The destination of the message is each station logged on with the usercode specified in the USERCODE field. The MAIL identifier is the usercode specified in the USERCODE field.

SUB_TYPE = 4

The DESTINATION field consists of a PROGRAM_NAME field. The destination of the message is each station attached to the program specified in the PROGRAM_NAME field. The MAIL identifier is specified by the program name in the PROGRAM_NAME field.

SUB_TYPE = 5

The DESTINATION field consists of a USER_IDENTIFIER field and a filler. The destination of the message is the MAIL file and the MAIL identifier is the specified by USER_IDENTIFIER.

SUB_VARIANT

The SUB_VARIANT field indicates whether or not the message is to be put in the MAIL file. The possible values for the SUB_VARIANT field are as follows:

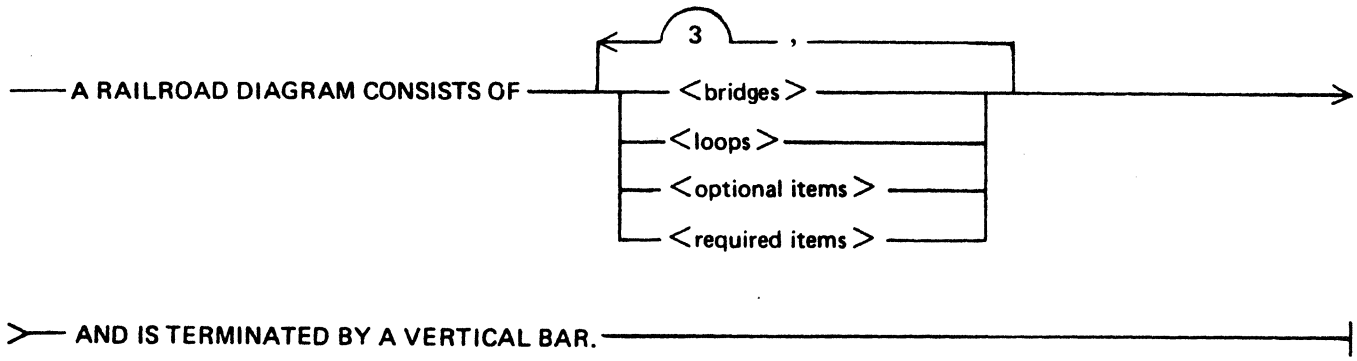
| SUB_VARIANT | Meaning |
|---|---|
| 0 | Send the message to the destination as specified in the SUB_TYPE field and put the message in the MAIL file. |
| 1 | Send the message to the destination only. |
| 2 | Put the message in the MAIL file only. |

# APPENDIX H
# SYNTAX AND NOTATION CONVENTIONS

Railroad diagrams show how syntactically valid statements can be constructed. Traversing a railroad diagram from left to right, or in the direction of the arrow heads, and adhering to the limits illustrated by bridges produces a syntactically valid statement. Continuation from one line of the diagram to another is represented by a right arrow (→) appearing at the end of the current line and the beginning of the next line. The complete syntax diagram is terminated by the vertical bar (|).

Items contained in broken brackets (< >) are syntactic variables which must be further defined or for which the user is required to supply the information requested.

Upper-case items must appear literally. Minimum abbreviations of upper-case items are underlined.




G50051

The following syntactically valid statements can be constructed from the above diagram:

A RAILROAD DIAGRAM CONSISTS OF <bridges> AND IS TERMINATED BY A VERTICAL BAR.

A RAILROAD DIAGRAM CONSISTS OF <optional items> AND IS TERMINATED BY A VERTICAL BAR.

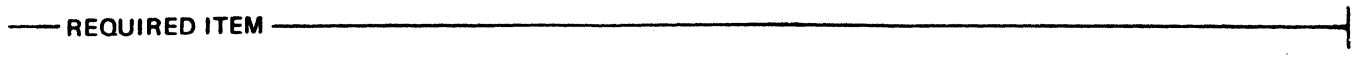
A RAILROAD DIAGRAM CONSISTS OF <bridges>, <loops> AND IS TERMINATED BY A VERTICAL BAR.

A RAILROAD DIAGRAM CONSISTS OF <optional items>, <required items>, <bridges>, <loops> AND IS TERMINATED BY A VERTICAL BAR.

## Required Items

No alternate path through the railroad diagram exists for required items or required punctuation.

Example:

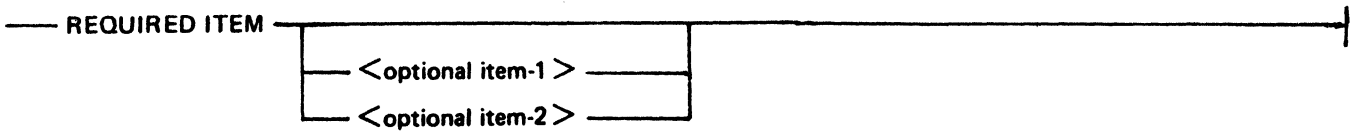



G50052

## Optional Items

A vertical list of items indicates that the user must make a choice of the items specified. An empty path through the list allows the optional item to be absent.

Example:




G50053

The following valid statements can be constructed from the above diagram:

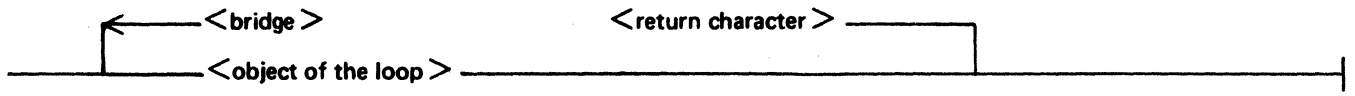
REQUIRED ITEM

REQUIRED ITEM <optional item-1>

REQUIRED ITEM <optional item-2>

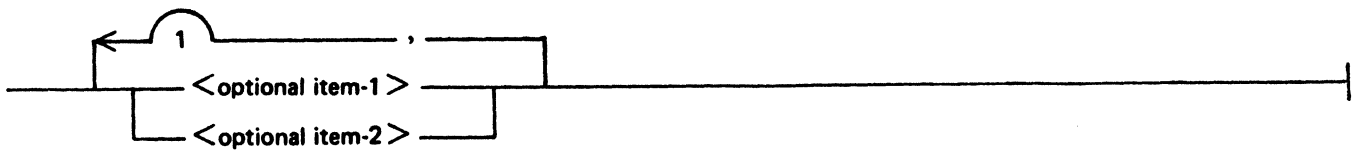
## Loops

A loop is a recurrent path through a railroad diagram and has the following general format.




G50054

Example:




G50055

The following statements can be constructed from the railroad diagram in the example.

<optional item-1>

<optional item-2>

<optional item-1>,<optional item-1>

<optional item-1>,<optional item-2>

<optional item-2>,<optional item-1>

<optional item-2>,<optional item-2>

A loop must be traversed in the direction of the arrow heads, and the limits specified by bridges cannot be exceeded.

## Bridges

A bridge illustrates the minimum or maximum number of times a path can be traversed in a railroad diagram.

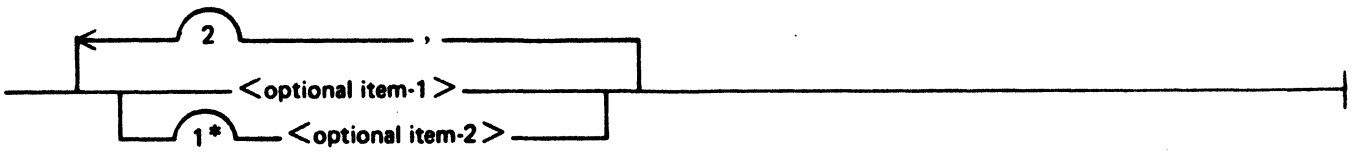
There are two forms of <bridges>.

<n> is an integer which specifies the maximum number of times the path can be traversed.

<n> is an integer which specifies the minimum number of times the path must be traversed.

G50056

Example:




G50057

The loop can be traversed a maximum of two times; however, the path for <optional item-2> must be traversed at least one time.

The following statements can be constructed from the railroad diagram in the example.

<optional item-1>,<optional item-2>

<optional item-2>,<optional item-2>,<optional item-1>

<optional item-2>

# INDEX

(SYSTEM)/USERCODE file 4-1, 4-6

? MCP control instructions 5-57

ATTACH command 5-3, 5-48
AUTO-START option 4-1, 4-3, 5-21

BROADCAST command 5-5
Burroughs Network Architecture (BNA) 2-4
    Host Services 2-4
    Network Services 2-4
BYE command 5-6, 5-34

CANDE 2-4, 4-6
CHANGE command 5-7
CLEAR command 5-10
CLOSE command 4-2, 5-11
CONTINUE command 5-12
COPIES option 4-3

DETACH command 5-13
DIRECTORY file 5-25, 6-1
DMPALL 2-3, 4-6
DUMP command 5-15
DYNAMIC command 5-16

EXCEPT option 4-3, 5-44

FILE command 5-17

HARDWARE command 5-18
HELP command 5-19

ID command 5-5, 5-20
INFO file 5-18, 5-33, 5-35
INITIATE command 4-1, 4-4, 5-21

JOBS command 5-22
JOBS file 2-3, 3-2, 4-1, 5-11, 5-21,
          5-34, 5-42, 5-44, 5-57

LOG-ON option 4-3
LOGON command 5-23

MACRO command 4-1, 5-23, 5-25, 6-1, 6-9
Macro procedures 4-1, 5-20, 5-23, 5-25
                 5-55, 5-56, 5-57, 6-1
                 6-3, 6-12
    creation 6-1
    invocation 4-1, 6-1
    invoking SMCS commands 6-3
    limits 6-9

# INDEX (Cont.)

MAIL commands 5-27
MAIL file 2-3, 5-27, E-1
MAKE command 5-28
MCP message routing 3-1
MCP name table 4-1, 4-11
MESSAGES option 4-3
MOVE command 5-31
MT 983 terminal 5-50
MT 985 terminal 5-50

Network controller 2-3, 4-1, 4-6, 4-11
    generation of 4-6
NEWS command 5-33
NO-EOF option 4-3, 5-34, 5-44
NO-RR option 4-4
NO-SCROLL option 4-4
NO-ZIP option 4-4

operating system (MCPII) 2-3

PASS command 4-1, 4-4, 5-21, 5-34,
    5-48, G-1
PASSWORD command 5-35
Program priorities 4-1, 4-6
Program switches 4-1
Program-pass input messages G-5
    message header format G-5
    remote key format G-5
    text format G-5
Program-pass messages G-1
Program-pass output messages G-1
    message header format G-2
    Protocol Type 0 G-3
    Protocol Type 1 G-4
    Protocol Type 2 G-4
    Protocol Type 3 G-4
    remote key format G-2
    text format G-3
RD 2-4, 4-3, 5-22
READY command 5-36
REMOVE command 5-37
REPORT command 5-38
Reverse scrolling 5-28
RR attribute 5-56, 5-57

SEND command 5-5, 5-40
Session number 5-56, 5-57
SIGN OFF command 4-3, 5-44
SIGN ON command 4-1, 4-4, 5-21, 5-42, 5-48

# INDEX (Cont.)

signal character 5-28, 5-29, 5-42, 5-44, 5-45, B-1
allowable B-1
initial value (*) 5-28
SIGNAL command 5-45
SMCS program capabilities 2-5
STOP command 5-6, 5-48
SYCOM 2-4, 2-5
SYSTEM command 5-49
SYSTEM/MAKEUSER 2-3

TABS command 5-50
TD 820 terminal 5-50
TD 830 terminal 5-50
TEXT/EDITOR 2-4, 2-5, 4-6
TRACE command 5-53, F-1
TRACE file 2-3, 5-18, F-1
Trace function 4-1
TRANSLATE command 5-54

USER command 5-5, 5-23, 5-48, 5-55
USER command 4-3
USERCODE option 4-4
USERFILE option 4-4

ZIP command 5-56

## Documentation Evaluation Form

Title: B 1000 Systems Supervisory Message Control System (SMCS) Reference Manual

Form No: 1108891

Date: May 1983

Burroughs Corporation is interested in receiving your comments and suggestions, regarding this manual. Comments will be utilized in ensuing revisions to improve this manual.

**Please check type of Suggestion:**

☐ Addition ☐ Deletion ☐ Revision ☐ Error

**Comments:**

**From:**

Name _____

Title _____

Company _____

Address _____

_____

Phone Number _____ Date _____

Remove form and mail to:

Burroughs Corporation
Corporate Documentation – West
1300 John Reed Court
City of Industry, CA 91745
U.S.A.